



# BMC Client Management - WebAPI Operations for Integrating with BCM

Version 12.2

# Legal Notices

©Copyright 2014 - 2016 BMC Software, Inc.

©Copyright 1994 - 2016 BladeLogic, Inc.

BMC, BMC Software, and the BMC Software logo and the BMC marks listed at [bmc.com/legal/trademarks](http://bmc.com/legal/trademarks) are the exclusive properties of BMC Software, Inc., or its affiliates or subsidiaries and are registered or are pending registration with the U.S. Patent and Trademark Office and/or in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

BladeLogic and the BladeLogic logo are the exclusive properties of BladeLogic, Inc. The BladeLogic trademark is registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BladeLogic trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

Cisco and Cisco NAC are registered trademarks or trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

IBM and IBM Domino are registered trademarks or trademarks of International Business Machines Corporation in the United States, other countries, or both.

IT Infrastructure Library® is a registered trade mark of AXELOS Limited.

Linux is the registered trademark of Linus Torvalds.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

UNIX is the registered trademark of The Open Group in the US and other countries.

BMC Software Confidential. BladeLogic Confidential.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

## **BMC Software Inc.**

2103 CityWest Blvd, Houston TX 77042-2828, USA

713 918 8800

Customer Support: 800 537 1813 (United States and Canada) or contact your local support center

# Contents

<b>REST.....</b>	<b>11</b>
Objects.....	11
Operations on Objects.....	11
GET Operations.....	12
/objects.....	12
Objects Return Classes.....	14
ObjectTypes.....	14
Object.....	15
Object Operations.....	15
GET Operations.....	16
/object/{objectId}/attrs.....	16
/object/{objectId}/insts.....	18
/object/{objectId}/inst/{instId}/attrs.....	22
PUT Operations.....	25
/object.....	25
/object/{objectId}/attr.....	28
/object/{objectId}/inst/{instId}/attrs.....	30
DELETE Operations.....	33
/object/{objectId}.....	33
/object/{objectId}/attr/{attrId}.....	34
Object Return Classes.....	36
DeleteObjectAttrResult.....	37
DeleteObjectResult.....	37
GetObjectAttrsResult.....	38
GetObjectInstAttrsResult.....	38
GetObjectInstsResult.....	38
ObjectAttr.....	38
ObjectInst.....	38
ObjectInstAttrs.....	39
ObjectResult.....	39
ObjectTypeAttr.....	39
PutObjectAttr.....	39
PutObjectAttrResult.....	39
PutObjectResult.....	40
Devices.....	40
Operations on Devices.....	40
GET Operations.....	41
/devices.....	41
Devices Return Classes.....	45
Device (devices).....	45
IndexContexts.....	46
DeviceInfo.....	46
Device API.....	47
Device Operations.....	47
GET Operations.....	49
/device/{deviceId}.....	49
/device/{deviceId}/financial.....	54
/device/{deviceId}/financial/summary.....	56
/device/{deviceId}/summary.....	59
/device/{deviceId}/inventory/software/summary.....	61
/device/{deviceId}/inventory/catalog/summary.....	63
/device/{deviceId}/inventory/hardware/summary.....	67
/device/{deviceId}/object/{objectId}/insts.....	71
/device/{deviceId}/windows/services.....	73
/device/{deviceId}/processes.....	76
/device/{deviceId}/windows/registry/keys.....	80
/device/{deviceId}/windows/registry/values.....	84

/device/{deviceId}/windows/event/categories.....	87
/device/{deviceId}/windows/events.....	91
/device/{deviceId}/filteredactions.....	97
/device/{deviceId}/oprule/history.....	100
/device/{deviceId}/jnl.....	102
/device/{deviceId}/connectivity.....	104
/device/{deviceId}/fs.....	106
/device/{deviceId}/fs/file.....	111
POST Operations.....	115
/device/{deviceId}/windows/service.....	115
/device/{deviceId}/ping.....	120
/device/{deviceId}/shutdown.....	122
/device/{deviceId}/wakeUp.....	125
/device/{deviceId}/audit.....	128
PUT Operations.....	130
/device/{deviceId}/process.....	130
/device.....	134
/device/{deviceId}.....	136
/device/{deviceId}/financial.....	140
/device/{deviceId}/windows/registry/key.....	143
/device/{deviceId}/windows/registry/value.....	147
/device/{deviceId}/inventory/patch/scan.....	151
/device/{deviceId}/object/{objectId}/inst.....	153
/device/{deviceId}/object/{objectId}/inst/{instId}/attrs.....	156
/device/{deviceId}/oprule/{opruleId}.....	161
/device/{deviceId}/session.....	164
/device/{deviceId}/fs/dir.....	167
/device/{deviceId}/fs/file.....	171
DELETE Operations.....	176
/device/{deviceId}/process/{processId}.....	176
/device/{deviceId}.....	180
/device/{deviceId}/windows/registry/key.....	182
/device/{deviceId}/windows/registry/value.....	186
/device/{deviceId}/object/{objectId}/inst/{instId}.....	191
/device/{deviceId}/fs/dir.....	195
/device/{deviceId}/fs/file.....	199
<i>Device Return Classes.....</i>	<i>203</i>
CatalogKeywordTranslation.....	206
DeviceAssetCatalog.....	206
DeviceAssetCatalogSummary.....	207
DeviceAssetFinancial.....	207
DeviceAssetFinancialSummary.....	208
DeviceAssetHardware.....	208
DeviceAssetHardwareSummary.....	209
DeviceAssetSoftware.....	210
DeviceAssetSoftwareSummary.....	210
DeviceAssetSummary.....	211
DeviceAssetSummaryResult.....	212
DeviceAttributes.....	212
DeviceAttributesInfo.....	216
DeviceCategoryResultValues.....	216
DeviceCategoryValues.....	216
DeviceCreate.....	217
DeviceData.....	217
DeviceDeleteDir.....	217
DeviceDeleteFile.....	217
DeviceExit.....	217
DeviceExitInfo.....	218
DeviceFileSystem.....	218
DeviceFileSystemDir.....	218
DeviceFileSystemDirInfo.....	219
DeviceFileSystemInfo.....	219
DeviceFinancialAttributes.....	219
DeviceInventoryAttrValue.....	222
DeviceInventoryAttrs.....	222
DeviceInventoryInsts.....	223
DeviceOperationResult.....	223
DeviceProcesses.....	223
DeviceProcessValues.....	223
DeviceReadFile.....	224

DeviceRegistryKey.....	225
DeviceRegistryKeys.....	225
DeviceRegistryPath.....	225
DeviceRegistryPathValue.....	225
DeviceRegistryValue.....	225
DeviceRegistryValues.....	226
DeviceResult.....	226
DeviceScan.....	226
DeviceScanResult.....	226
DeviceService.....	227
DeviceServiceInfo.....	227
DeviceTerminateResult.....	227
DeviceUpdate.....	227
DeviceWindowsEvent.....	228
DeviceWindowsEventResult.....	228
DirectoryInfo.....	229
FileInfo.....	229
FilteredActionsResult.....	229
FilteredActionsValues.....	230
HardwareKeywordTranslation.....	231
InventoryInstCreate.....	232
JsonResult.....	232
KeywordTranslation.....	232
OpRuleHistoryResult.....	233
SessionOperationResult.....	233
SessionResult.....	234
SoftwareKeywordTranslation.....	234
<b>Group.....</b>	<b>234</b>
<i>Group Operations.....</i>	<i>234</i>
GET Operations.....	235
/group/{groupid}/hierarchy.....	235
/group/subgroups.....	238
/group/subgroup/{groupid}/subgroups.....	240
/group/devices.....	243
/group/subgroup/{groupid}/devices.....	244
/user/subgroups.....	247
/user/group/{groupid}/subgroups.....	249
/user/users.....	252
/user/group/{groupid}/users.....	254
/package/subfolders.....	257
/package/folder/{folderId}/subfolders.....	259
/package/packages.....	261
/package/folder/{folderId}/packages.....	263
POST Operations.....	266
/group/{groupIds}/wakeUp.....	266
<i>Group Return Classes.....</i>	<i>268</i>
DeviceGroups.....	268
Devices.....	269
GroupDevices.....	269
GroupOperationResult.....	270
GroupPackages.....	270
GroupUsers.....	270
GroupValues.....	270
GroupValuesInfo.....	271
Packages.....	271
PackagesFolders.....	272
UserGroups.....	272
Users.....	272
<b>Operational Rule.....</b>	<b>273</b>
<i>Operational Rule Operations.....</i>	<i>273</i>
GET Operations.....	276
/oprule/subfolders.....	276
/oprule/folder/{folderId}/subfolders.....	278
/oprule/folder/{folderId}.....	281
/oprule/rules.....	283
/oprule/rule/{ruleId}.....	286
/oprule/folder/{folderId}/rules.....	288
/oprule/rule/{ruleId}/step/assignments.....	291
/oprule/rule/{ruleId}/step/assignment/{assignmentId}.....	294

/oprule/icons.....	298
/oprule/step/classes.....	299
/oprule/steps.....	300
/oprule/step/{stepTypeId}.....	302
/oprule/rule/{ruleId}/device/assignments.....	305
/oprule/rule/device/assignment/{assignmentId}.....	307
/oprule/rule/{ruleId}/device/group/assignments.....	310
/oprule/rule/device/group/assignment/{assignmentId}.....	312
/oprule/rule/{ruleId}/user/assignments.....	314
/oprule/rule/user/assignment/{assignmentId}.....	316
/oprule/rule/{ruleId}/user/group/assignments.....	318
/oprule/rule/user/group/assignment/{assignmentId}.....	320
/oprule/rule/{ruleId}/dependencies.....	322
/oprule/rule/{ruleId}/packages.....	324
/schedule/{scheduleId}.....	326
PUT Operations.....	329
/oprule/folder/{folderId}/subfolder.....	329
/oprule/rule.....	331
/oprule/folder/{folderId}/rule.....	334
/oprule/folder/{folderId}.....	338
/oprule/rule/{ruleId}.....	340
/oprule/rule/{ruleId}/step/{stepTypeId}.....	343
/oprule/rule/{ruleId}/device/{deviceId}.....	347
/oprule/rule/{ruleId}/device/group/{groupId}.....	351
/oprule/rule/{ruleId}/user/group/{groupId}.....	354
/oprule/rule/{ruleId}/user/{userId}.....	357
/oprule/rule/{ruleId}/step/assignment/{assignmentId}.....	360
/oprule/rule/{ruleId}/device/{deviceId}.....	363
/oprule/rule/{ruleId}/device/group/{groupId}.....	367
/oprule/rule/user/assignment/{assignmentId}.....	370
/oprule/rule/user/group/assignment/{assignmentId}.....	372
/oprule/rule/{ruleId}/dependency/{dependentRuleId}.....	374
/oprule/rule/{ruleId}/package/{packageId}.....	378
/schedule/{scheduleId}.....	380
DELETE Operations.....	382
/oprule/folder/{folderId}.....	382
/oprule/rule/{ruleId}/step/assignment/{assignmentId}.....	384
/oprule/rule/{ruleId}.....	387
/oprule/rule/device/assignment/{assignmentId}.....	389
/oprule/rule/device/group/assignment/{assignmentId}.....	392
/oprule/rule/user/assignment/{assignmentId}.....	394
/oprule/rule/user/group/assignment/{assignmentId}.....	396
/oprule/rule/{ruleId}/dependency/{dependentRuleId}.....	398
/oprule/rule/{ruleId}/package/{packageId}.....	400
Operational Rule Return Classes.....	402
AssignmentDetails.....	406
AssignmentId.....	406
AssignmentValuesResult.....	406
Device (operational rule).....	406
DeviceAssignment.....	407
DeviceAssignmentAttrs.....	408
DeviceAssignmentAttrsArray.....	410
DeviceAssignmentAttrsObject.....	410
DeviceAssignmentObj.....	410
DeviceGroup.....	413
DeviceGroupAssignment.....	413
DeviceGroupAssignmentAttrs.....	414
DeviceGroupAssignmentAttrsArray.....	416
DeviceGroupAssignmentAttrsObject.....	417
DeviceGroupAssignmentObj.....	417
ExecutionError.....	418
FolderAttrs.....	419
FolderValuesResult.....	419
MyAppIcons.....	419
OperationalRuleArray.....	420
OperationalRuleAttrs.....	420
OperationalRuleAttrsForUpdate.....	421
OperationalRuleFolderArray.....	421
OperationalRuleFolderValues.....	421
OperationalRuleFolderValuesArray.....	421

OperationalRuleValues.....	422
OperationalRuleValuesArray.....	422
OperationalRuleValuesResult.....	424
OperationResult.....	424
OpRuleDependencies.....	424
OpRuleDependencyDetails.....	424
OpRuleDependencyOrder.....	424
OpRuleDependencyPosition.....	425
OpRuleDependencyResult.....	425
PackageAssignment.....	425
PackageAssignmentDetails.....	425
PutRuleFolderResult.....	425
PutRuleResult.....	426
PutRuleStepAssignmentResult.....	426
Schedule.....	426
ScheduleCron.....	426
ScheduleDetails.....	427
ScheduleFrequency.....	427
ScheduleId.....	429
ScheduleResult.....	429
ScheduleValidity.....	429
StepAssignmentArray.....	430
StepAssignmentForUpdate.....	430
StepAttrs.....	430
StepAttrsResult.....	432
StepClasses.....	432
StepDetails.....	432
StepList.....	432
StepParamAttrs.....	432
StepParameters.....	433
StepParameterValues.....	433
StepParamValues.....	433
Steps.....	434
User.....	434
UserAssignment.....	434
UserAssignmentAttrs.....	436
UserAssignmentAttrsArray.....	438
UserAssignmentAttrsObject.....	438
UserAssignmentAttrsPut.....	438
UserAssignmentObj.....	439
UserGroup.....	441
UserGroupAssignment.....	441
UserGroupAssignmentAttrs.....	443
UserGroupAssignmentAttrsArray.....	445
UserGroupAssignmentAttrsObject.....	445
UserGroupAssignmentObj.....	445
<b>Reports.....</b>	<b>447</b>
<i>Operations on Reports.....</i>	<i>448</i>
GET Operations.....	448
/reports/folders.....	448
/reports/folder/{folderId} .....	450
/reports/reports.....	453
/reports/folder/{folderId}/reports .....	456
/reports/public.....	459
<i>Reports Return Classes.....</i>	<i>462</i>
ReportArray.....	463
ReportFolderArray.....	463
ReportFolderInfo.....	463
ReportInfo.....	464
ReportsPublic.....	464
ReportsPublicValues.....	464
<b>Report.....</b>	<b>465</b>
<i>Report Operations.....</i>	<i>465</i>
GET Operations.....	466
/report/{reportId}/generate.....	466
/report/{reportId}/load.....	467
<i>Report Return Classes.....</i>	<i>469</i>
ReportGenerate.....	469
ReportLoad.....	469

ReportLoadReportData.....	470
ReportLoadSubReportData.....	471
History.....	471
<i>Logging History Operations.....</i>	471
GET Operations.....	472
/history/contexts.....	472
DELETE Operations.....	473
/history/context/{contextId}.....	473
<i>History Return Classes.....</i>	475
ContextInfo.....	475
HistoryOperationResult.....	475
RequestContext.....	475
Cmdb.....	476
<i>CMDB Database Operations.....</i>	476
GET Operations.....	477
/cmdb/import.....	477
POST Operations.....	479
/cmdb/import.....	479
DELETE Operations.....	480
/cmdb/datafile.....	480
<i>Database Return Classes.....</i>	482
CmdbDataResult.....	482
CmdbExportResult.....	482
CmdbObject.....	483
CmdbOperationResult.....	483
DeviceType.....	483
GroupID.....	484
Localization (I18N).....	484
<i>Localization (I18N) Operations.....</i>	484
GET Operations.....	485
/i18n/error/{errorId}.....	485
POST Operations.....	486
/i18n/keywords.....	486
<i>Localization Return Classes.....</i>	488
ErrorCodeText.....	488
KeywordsTranslation.....	488
KeywordsType.....	488
Enums.....	489
<i>Enumerations.....</i>	489
GET Operations.....	489
/enum/groups.....	489
/enum/group.....	490
<i>Enumeration Return Classes.....</i>	492
Enumerations.....	493
EnumGroupDetails.....	493
EnumGroups.....	493
EnumMembers.....	493
Discovery.....	494
<i>Discovery Operations.....</i>	494
Discovery GET Operations.....	496
/discovery/target/lists/subfolders.....	496
/discovery/target/lists/folder/{folderId}/subfolders.....	498
/discovery/target/lists/folder/{folderId}.....	501
/discovery/target/lists.....	504
/rollout/configuration/{configurationId}.....	506
/discovery/scan/configurations/subfolders.....	508
/discovery/scan/configurations/folder/{folderId}/subfolders.....	509
/discovery/scan/configurations/folder/{folderId}.....	512
/discovery/scan/configurations.....	515
/discovery/scan/configuration/{configurationId}/protocols.....	517
/discovery/device/{deviceId}/scans/subfolders.....	519
/discovery/scans/folder/{folderId}/subfolders.....	522
/discovery/device/{deviceId}/scans.....	525
/discovery/scans/folder/{folderId}.....	528
/discovery/scan/{scanId}/configuration.....	531
/discovery/scan/{scanId}/target/lists.....	532
/discovery/scan/{scanId}/assignment.....	534
Discovery POST Operations.....	536



/discovery/scan/assignment/{assignmentId}.....	536
Discovery PUT Operations.....	539
/discovery/target/lists/subfolder.....	539
/discovery/target/lists/folder/{folderId}/subfolder.....	541
/discovery/target/list.....	544
/discovery/target/list/folder/{folderId}.....	546
/discovery/target/list/{targetListId}/members.....	549
/discovery/scan/configurations/subfolder.....	552
/discovery/scan/configurations/folder/{folderId}/subfolder.....	554
/discovery/scan/configuration.....	557
/discovery/scan/configuration/folder/{folderId}.....	559
/discovery/scan/configuration/{configurationId}/protocol/{assignmentId}.....	562
/discovery/scan/configuration/{configurationId}/protocols.....	566
/discovery/device/{deviceId}/scans/subfolder.....	569
/discovery/scans/folder/{folderId}/subfolder.....	572
/discovery/device/{deviceId}/scan.....	576
/discovery/scan/folder/{folderId}.....	579
/discovery/scan/{scanId}/configuration/{configurationId}.....	582
/discovery/scan/{scanId}/target/list/{targetListId}.....	585
/discovery/scan/{scanId}/configuration/{configurationId}/target/list/{targetListId}.....	588
Discovery DELETE Operations.....	592
/discovery/target/lists/folder/{folderId}.....	592
/discovery/target/list/{targetListId}.....	594
/discovery/target/list/{targetListId}/members/{assignmentId}.....	596
/discovery/scan/configurations/folder/{folderId}.....	599
/discovery/scan/configuration/{configurationId}.....	601
/discovery/scans/folder/{folderId}.....	603
/discovery/scan/{scanId}.....	605
/discovery/scan/{scanId}/configuration/{configurationId}.....	607
/discovery/scan/{scanId}/target/list/{targetListId}.....	610
Discovery Return Classes.....	613
AllProtocolAttrs.....	615
AssignValuesResult.....	616
CredentialAttrs.....	616
FolderAttrs.....	616
FolderValuesResult.....	616
MembersAttrsArray.....	617
MemberAttrs.....	617
ObjectFolderArray.....	617
ObjectFolderValuesArray.....	618
ObjectFolderValues.....	618
OperationResult.....	618
ProtocolAssignValuesResult.....	619
ProtocolAttrs.....	619
PutObjectFolderResult.....	619
PutScanConfigurationResult.....	619
PutTargetListResult.....	619
ScanAssignment.....	620
ScanAttrs.....	620
ScanConfigurationArray.....	620
ScanConfigurationAssign.....	621
ScanConfigurationAttrs.....	621
ScanConfigurationListValues.....	621
ScanConfigurationProtocolArray.....	621
ScanConfigurationProtocolValues.....	622
ScanConfigurationProtocolValuesArray.....	622
ScanConfigurationValuesArray.....	622
ScanConfigurationValuesResult.....	623
ScanResult.....	623
SNMPCredentialAttrs.....	623
SNMPProtocolAttrs.....	624
StatusValues.....	624
TargetAssignIds.....	624
TargetListArray.....	625
TargetListAttrs.....	625
TargetListMemberArray.....	626
TargetListMemberValues.....	626
TargetListMemberValuesArray.....	626
TargetListScanAssignArray.....	627
TargetListScanAssignValues.....	627

TargetListScanAssignValuesArray.....	627
TargetListValues.....	628
TargetListValuesArray.....	628
TargetListValuesResult.....	628
Practical Examples.....	628
<i>Creating and Using a new Custom Inventory Object Type.....</i>	<i>629</i>
<i>Creating and Using a new Quick Link Operational Rule.....</i>	<i>633</i>
<i>Creating a new Operational Rule in a new Folder.....</i>	<i>635</i>
<i>Creating Software Distribution Rule.....</i>	<i>639</i>
<i>Creating a Dependency between Operational Rules.....</i>	<i>644</i>
<i>Assigning a Specific Schedule to an Operational Rule.....</i>	<i>645</i>
Appendix.....	648
<i>Most Common Object Types and their IDs.....</i>	<i>648</i>

# REST

As REST is only an architectural style and not a protocol for example, there is no "official" standard for REST web services. However, it can use standards like HTTP, URI, XMS, and so on, like any standard.

The REST web API is a web service activated via a module in BMC Client Management using HTTP and a collection of resources, with four defined aspects:

- the base URI for the web service, such as `http(s)://localhost:1611/wsdoc`.
- the Internet media type of the data supported by the web service. This is often XML but can be any other valid Internet media type provided that it is a valid hypertext standard.
- the set of operations supported by the web service using the HTTP methods GET, PUT, POST, and DELETE.
- The API must be hypertext driven.

The following table shows how the HTTP methods are used:

- GET: List the URIs and perhaps other details of the collection's members if executed on a collection URI, or retrieve a representation of the addressed member of the collection, expressed in an appropriate Internet media type if executed on an individual element.
- POST: Create a new entry in the collection. The new entry's URI is assigned automatically and is usually returned by the operation. This method is mainly used on a collection URI not for execution on an individual element. If run on an individual element it treats the addressed member as a collection in its own right and creates a new entry in it.
- PUT: Replace the entire collection with another collection if executed on a collection URI. If executed on an individual element it replaces the addressed member of the collection, or if it doesn't exist, creates it.
- DELETE: Delete the entire collection or the addressed member of the collection.

The following sections detail all REST operations and classes that were implemented in the BCM WebApi module for interaction with the CM database. The following classes are available:

- [Objects](#)
- [Object](#)
- [Devices](#)
- [Device](#)
- [Device Group](#)
- [Operational Rule](#)
- [Reports](#)
- [Report](#)
- [History](#)
- [Cmdb](#)
- [i18n](#)
- [enum](#)
- [history](#)

## Objects

The operations of this group provide you the possibility to execute a search on different object types of BCM.

[Objects Operations](#)

[Return Classes](#)

## Operations on Objects

The following operations are available on objects:

## GET

The following get operations are available:

Action	Description
<a href="#">/objects</a>	Get the list of BCM object types by class.

## GET Operations

### /objects

This operation returns the list of BCM object types for a specific class. For example, you can find all object types that are currently available for the custom inventory by specifying the value `_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_` (Custom Inventory).

### Syntax

GET `/objects?class=ObjectTypeClass`

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
class	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The object class for which to return the available object types, which is <code>_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_</code> for custom inventory.

## Returns

Returns the class List[[ObjectTypes](#)].

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16010	Login failed.

## Example

The following example shows the request and output for the custom inventory for a default installation on which no new objects are created yet.

### Request URL

```
https://localhost:1611/api/1/objects?class=_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_
```

### Response Body

```
{
  "types": [
    {
      "id": 6109,
      "name": "Windows MSS",
      "class": "_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_",
      "smallIcon": null,
      "largeIcon": null,
      "parentType": null,
      "availableForSearch": false,
      "availableForQuery": false,
      "isGroup": false,
      "checkSecurity": false
    },
    {
      "id": 6110,
      "name": "Windows Security Settings",
      "class": "_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_",
      "smallIcon": null,
      "largeIcon": null,
      "parentType": null,
      "availableForSearch": false,
      "availableForQuery": false,
      "isGroup": false,
      "checkSecurity": false
    }
  ]
}
```

## Objects Return Classes

The following classes can be returned by the different objects actions:

Class	Description
<i>ObjectTypes</i>	This class returns the information available for a specific object type.

### ObjectTypes

This class returns the information available for a specific object type.

Parameter	Type	Description
id	integer	The unique BCM database identifier of the type.
<b>Name</b>	string	The name of the type, which can be any one of the types known in the BCM database, such as <code>_DB_OBJECTTYPE_DEVICE_</code> , <code>_DB_OBJECTTYPE_DEVICEGROUP_</code> , <code>_DB_OBJECTTYPE_ADMINISTRATOR_</code> , and so on.
class	string	The object type class, which can be either <code>_DB_OBJECTTYPCCLASS_BASIC_</code> for the general asset summary, <code>_DB_OBJECTTYPCCLASS_SOFTWAREINVENTORY_</code> for software inventory, <code>_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_</code> for hardware inventory, <code>_DB_OBJECTTYPCCLASS_CUSTOMINVENTORY_</code> for custom inventory, <code>_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_</code> for the security inventory, <code>_DB_OBJECTTYPCCLASS_POWERMANAGEMENTINVENTORY_</code> for the power management inventory, <code>_DB_OBJECTTYPCCLASS_CONNECTIVITYINVENTORY_</code> for the connectivity inventory or <code>_DB_OBJECTTYPCCLASS_VIRTUALINFRASTRUCTUREINVENTORY_</code> for the inventory of virtual devices.
smallIcon	string	The name and relative path of the small icon file to be used to represent this device.
largeIcon	string	The name and relative path of the large icon file to be used to represent this device.
ParentType	string	Specifies if the object type has a parent, such as individual devices or queries, and if yes which. For example, for a <code>_DB_OBJECTTYPE_DEVICE_</code> the parent type will always be <code>_DB_OBJECTTYPE_DEVICEGROUP_</code> , for an <code>_DB_OBJECTTYPE_ADMINISTRATOR_</code> the parent type is <code>_DB_OBJECTTYPE_ADMINISTRATORGROUP_</code> . If a type is already a parent type this parameter remains empty (null). Also if the object type is the access to a functionality it does not have a parent.
availableForSearch	boolean	Specifies if the object type can be searched on in the BCM Search functionality.
availableForQuery	boolean	Specifies if the object type can be used as a criterion in BCM queries.

Parameter	Type	Description
isGroup	boolean	Defines if the object type is a parent, that is, a group or a folder, in which case this value is "true", otherwise it is "false".
checkSecurity	boolean	Defines if the new object type is to have a <b>Security</b> tab.

## Object

The operations of this group provide you the possibility to execute different types of actions on one specific object type.

[Object Operations](#)

[Return Classes](#)

## Object Operations

The following operations are available for the object object:

### GET

The following get operations are available:

Action	Description
<a href="#">/object/{objectId}/attrs</a>	Get an object type attribute list
<a href="#">/object/{objectId}/insts</a>	Get an object type instance list
<a href="#">/object/{objectId}/inst/{instId}/attrs</a>	Get an object type instance attribute list

### PUT

The following put operations are available:

Action	Description
<a href="#">/object</a>	Create a new object type
<a href="#">/object/{objectId}/attr</a>	Create a new object type attribute
<a href="#">/object/{objectId}/inst/{instId}/attrs</a>	Set an object type instance attribute list

### DELETE

The following put operations are available:

Action	Description
<a href="#">/object/{objectId}</a>	Delete an object

Action	Description
<a href="#">/object/{objectId}/attr/{attrId}</a>	Delete an object type attribute

GET Operations

**/object/{objectId}/attrs**

This operation returns the list of all attributes of a specific object type.

Syntax

GET /object/{objectId}/attrs

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object. This value is returned by the <code>GET /objects</code> operation.

Returns

Returns the class `List[GetObjectAttrsResult]`.



## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.

## Example

The following example shows the list of attributes that exist for object 6110 (custom inventory object type called "Windows MSS").

### Request URL

```
https://localhost:1611/api/1/object/6110/attrs
```

### Response Body

```
{
  "attrs": [
    {
      "name": "_DB_ATTR_LASTINSERT_",
      "column": "CreateTime",
      "type": "Date",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 0,
      "displayInList": false,
      "needTranslation": false,
      "valueCheckRegex": null,
      "id": 597051
    },
    {
      "name": "_DB_ATTR_LASTUPDATE_",
      "column": "ModifyTime",
      "type": "Date",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 0,
      "displayInList": false,
      "needTranslation": false,
      "valueCheckRegex": null,
      "id": 597052
    },
    {
      "name": "InstanceName",
      "column": "InstanceName",
      "type": "String",
      "enumName": null,

```

```

    "units": null,
    "readOnly": false,
    "displayOrder": 1,
    "displayInList": false,
    "needTranslation": false,
    "valueCheckRegex": null,
    "id": 597050
  },
  {
    "name": "EntryValue",
    "column": "ATTR_597053",
    "type": "String",
    "enumName": null,
    "units": null,
    "readOnly": false,
    "displayOrder": 2,
    "displayInList": false,
    "needTranslation": false,
    "valueCheckRegex": null,
    "id": 597053
  }
]
}

```

### **/object/{objectId}/insts**

With this operation you can find all instances of a specific object type.

### **Syntax**

GET /object/{objectId}/insts?

source=HierarchySource&attr=AttrToFind&operator=OperatorToUse&value?StringToFind

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the object.
source	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	List	Select the object under which to execute the search.

Name	Type	Mand.	Data Type	Description
attr	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The attribute name to be searched. Specific attribute names can be found via the GET <i>/object/{objectId}/attrs</i> operation.
operator	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	List	The search operator to be applied, which can be either <i>_DB_QUERYOPERATOR_CONTAINS_</i> if the search result is only to contain the value entered into the <i>attr</i> field or <i>_DB_QUERYOPERATOR_EQUALS_</i> if it is to be an exact match of this value.

Name	Type	Mand.	Data Type	Description
value	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The corresponding attribute value that is to be found.

## Returns

Returns the class [GetObjectInstsResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.

## Example

The following example searches in the hierarchy for devices containing the capital letter *B* in their name.

### Request URL

```
https://localhost:1611/api/1/object/100/insts?  
source=Hierarchy&attr=_DB_ATTR_NAME_&operator=_DB_QUERYOPERATOR_CONTAINS_&value=B
```

### Response Body

```
{  
  "insts": [  
    {  
      "id": 1000,  
      "name": "Device Groups/All Devices/BMC-CB7L5S1",  
      "smallIcon": "device/master_green_win.png"  
    }  
  ]  
}
```

## /object/{objectId}/inst/{instId}/attrs

This operation returns the list of all attributes of an object type instance.

## Syntax

GET /object/{objectId}/inst/{instId}/attrs

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object.
instanceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type instance.

## Returns

Returns the class [GetObjectInstAttrsResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
10	The requested object could not be found.
13	Incorrect parameter. Please verify the supplied parameters.
16010	Login failed.
16040	Bad object type.

## Example

The following example searches the list of attributes of a device (type id 100) and specifically for the device with the ID 1000 (master).

### Request URL

```
https://localhost:1611/api/1/object/100/inst/1000/attrs
```

### Response Body

```
{
  "Values": [
    {
      "_DB_ATTR_OSD_PARENT_": 0,
      "_DB_ATTR_SMALLICON_": "device/master_green_win.png",
      "_DB_ATTR_LARGEICON_": "device/master_green_win_l.png",
      "_DB_ATTR_ENTRYDN_": "",
      "_DB_ATTR_DEV_GUID_": "00016FAAC15D936664E1EBF0B1A440DD1DE4",
      "_DB_ATTR_DEV_RETIREDBY_": "",
      "_DB_ATTR_DEVICE_FAMASSETTAG_": "",
      "_DB_ATTR_DEVICE_PRIMARYUSER_": "ADPROD//sertle",
      "_DB_ATTR_NAME_": "BMC-CB7L5S1",
      "UserName": "ADPROD//sertle",
      "FAMLocation": "",
      "_DB_ATTR_DEV_UPDATETIME_": "2013/03/04 07:52:45",
      "_DB_ATTR_DEV_TYPE_": "laptop",
      "_DB_ATTR_DEVICE_IPADDRESS_": "10.5.159.27",
      "_DB_ATTR_DEVICE_TOPOLOGYTYPE_": "_DB_DEVTYPE_MASTER_",
      "_DB_ATTR_DEV_DOMAINNAME_": "BMC-CB7L5S1.adprod.bmc.com",
      "_DB_ATTR_DEV_OSNAME_": "Microsoft Windows 7 Enterprise Edition 64-bit",
      "_DB_ATTR_DEVICE_HOSTID_": "CB7L5S1",
      "_DB_ATTR_PARENT_": "",
      "_DB_ATTR_VIMVENDOR_": "NotVirtual",
      "_DB_ATTR_HYPERVISORVENDOR_": "NotHypervisor",
      "_DB_ATTR_HYPERVISORVERSION_": "",
      "_DB_ATTR_DEV_NETWORKNAME_": "10.5.159.27",
      "_DB_ATTR_DEVICE_NETBIOSNAME_": "BMC-CB7L5S1",
    }
  ]
}
```



```

    "_DB_ATTR_DEVICE_NETWORKMASK_": "255.255.252.0",
    "_DB_ATTR_DEVICE_MACADDRESS_": "24:77:03:45:AA:18",
    "_DB_ATTR_DEVICE_DISKSERNUM_": "90bf9a2e",
    "_DB_ATTR_DEVICE_UNDERNAT_": false,
    "_DB_ATTR_VPROAVAILABLE_": false,
    "_DB_ATTR_DEV_HTTPPORT_": 1610,
    "_DB_ATTR_DEV_HTTPCONSOLEPORT_": 1611,
    "_DB_ATTR_DEV_SSL_": "_DB_SSL_HTTPS_SENDANDRECEIVE_",
    "_DB_ATTR_DEV_AGENTVERMAJOR_": 12,
    "_DB_ATTR_DEV_AGENTVERMINOR_": 0,
    "_DB_ATTR_DEV_AGENTREVISION_": "0",
    "_DB_ATTR_DEV_AGENTBUILD_": "130221s",
    "_DB_ATTR_PATCHCONFIGFILEVERSION_": "2.0.0.3712",
    "_DB_ATTR_DEV_OSVERMAJOR_": 6,
    "_DB_ATTR_DEV_OSVERMINOR_": 1,
    "_DB_ATTR_DEV_OSREVISION_": "",
    "_DB_ATTR_DEV_OSBUILD_": "7600",
    "_DB_ATTR_PKGFACTORY_": true,
    "_DB_ATTR_PATCHPROXY_": true,
    "_DB_ATTR_OSD_": false,
    "_DB_ATTR_RISCANNER_": true,
    "_DB_ATTR_ROLLOUTSRV_": true,
    "_DB_ATTR_RESTROLE_": false,
    "_DB_ATTR_CREATEDBY_": "system",
    "_DB_ATTR_CREATETIME_": "2013/02/22 10:02:03",
    "_DB_ATTR_LASTMODIFIEDBY_": "",
    "_DB_ATTR_LASTMODIFYTIME_": "",
    "_DB_ATTR_NOTES_": ""
  }
],
"ErrorCode": 0
}

```

## PUT Operations

### /object

This operation allows you to create new object types, for example to add a specific custom inventory object type for your inventory needs.

### Syntax

PUT /object?class=ObjectClass&type=MyNewObjectType

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
class	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	LIST	Inventory class for new object type, <code>_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_</code> for the custom inventory.
type	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The name of the new object type to be created.

## Returns

Returns the class *PutObjectResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.

## Example

This following example creates the new custom inventory object type of name *MyNewObjectType*.

### Request URL

`https://localhost:1611/api/1/object?class=_DB_OBJECTTYPCCLASS_CUSTOMINVENTORY_&type=MyNewObjectType`

### Response Body

```
{
  "Values": {
    "ObjectTypeId": 6114
  },
  "ErrorCode": 0
}
```

whereby `ObjectTypeId` the object ID of the new object type is, which is requested in other operations such as `PUT /object/{objectId}/attr`, to create attributes for this object type or `DELETE /object/{objectId}` when deleting the object type.

**/object/{objectId}/attr**

With this operation you can add new attributes to an existing object type, for example the attributes of a custom inventory object type.

**Syntax**

PUT /object/{objectId}/attr

The body must contain the definitions of the PutObjectAttr class.

```
{ "AttrName": "NameOfNewAttr", "AttrType": "AttrDataType" }
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type for which the attributes are to be created. This ID is returned either by the GET /objects or the PUT /object operation.

Name	Type	Mand.	Data Type	Description
PutObjectAttr	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	<i>PutObjectAttr</i>	The list of attributes to set with their desired values.

## Returns

Returns the class *PutObjectAttrResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.

Error Code	Description
16002	SQL execution error
16010	Login failed.
16040	Bad object type.

## Example

The following example creates a new attribute (`MyNewAttr`) for an custom inventory object type called `MyNewObjectType` (ID 6114).

### Request URL

```
https://localhost:1611/api/1/object/6114/attr
```

whereby the body contains the following information:

```
{
  "AttrName": "MyNewAttr",
  "AttrType": "string"
}
```

### Response Body

```
{
  "ObjectId": 6114,
  "AttrName": "MyNewAttr",
  "AttrType": "string",
  "ErrorCode": 0
}
```

## /object/{objectId}/inst/{instId}/attrs

This operation allows you to set one or more attributes of an object type instance.

## Syntax

```
PUT /object/{objectId}/inst/{instId}/attrs
```

The body must contain the definitions of the *ObjectTypeAttr* class.

```
{ "AttrName": "string|int|boolean" }
```



### Note

The keyword string can contain more than one value, the values must be separated with a colon (,).

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type for which the instance is to be created.
instanceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type instance which is to be created.

Name	Type	Mand.	Data Type	Description
ObjectTypeAttr	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	<i>ObjectTypeAttr</i>	The object attributes with their values that are to be set.

## Returns

Returns the class *OperationResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error



Error Code	Description
16010	Login failed.
16040	Bad object type.

## DELETE Operations

### /object/{objectId}

This operation allows you to delete a specific object type, for example a custom inventory object type.

### Syntax

DELETE /object/{objectId}

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type to be deleted. You can find the ID of a specific object type by executing the GET /objects operation.

### Returns

Returns the class [DeleteObjectResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16010	Login failed.
16047	This object is used by a query and cannot be deleted.

## Example

This example deletes the newly created custom inventory object type `MyNewObjectType`.

### Request URL

```
https://localhost:1611/api/1/object/6114
```

### Response Body

```
{
  "ErrorCode": 0
}
```

## /object/{objectId}/attr/{attrId}

This operation allows you to delete a specific attribute of an object type, such as for example an attribute that was part of a custom inventory object type.

## Syntax

```
DELETE /object/{objectId}/attr/{attrId}
```

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type. You can find the ID of a specific object by executing the <code>GET /objects</code> operation.
attrId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type attribute to be deleted. The IDs of specific object types are returned by the <code>GET /object/{objectId}/attrs</code> operation.

## Returns

Returns the class *DeleteObjectAttrResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.
16043	The supplied attribute name does not exist.
16047	This object is used by a query and cannot be deleted.
16048	Mandatory attribute that cannot be deleted.

## Example

This example deletes the newly created attribute `MyNewAttr` of the as well newly created custom inventory object type `MyNewObjectType`.

### Request URL

```
https://localhost:1611/api/1/object/6114/attr/597096
```

### Response Body

```
{
  "ObjectId": 6114,
  "AttrId": 597096,
  "ErrorCode": 0
}
```

## Object Return Classes

The following classes can be returned by the different object actions:

Class	Description
<i>DeleteObjectAttrResult</i>	The delete operation result of an object attribute.
<i>DeleteObjectResult</i>	The delete operation result of an object type
<i>GetObjectAttrsResult</i>	Get the object type attribute list
<i>GetObjectInstAttrsResult</i>	Return the object type instance attribute list.
<i>GetObjectInstsResult</i>	Get the object type's instances
<i>ObjectAttr</i>	Object type attribute with all its parameters
<i>ObjectInst</i>	Get the object type instance
<i>ObjectInstAttrs</i>	Return the list of attributes of the object type instance.
<i>ObjectResult</i>	Return the unique ID of the new object type.
<i>ObjectTypeAttr</i>	Provides the list of attributes and their values to set.
<i>PutObjectAttr</i>	Provide the attribute parameters and their values.
<i>PutObjectAttrResult</i>	Return the result of the object type attribute create operation.
<i>PutObjectResult</i>	Return if the creation operation was successful.

### DeleteObjectAttrResult

This class returns the result of the delete operation of an object attribute. It displays the following information:

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.
AttrId	integer	The unique BCM database identifier of the attribute that was deleted.
ObjectTypeId	integer	The unique BCM database identifier of the object type of which the attribute that was deleted.

### DeleteObjectResult

This class returns the result of the delete operation of an object type. It displays the following information:

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### GetObjectAttrsResult

This class returns the list of object attributes of a specific object type. It displays the following information:

Parameter	Type	Description
attrs	List[ <a href="#">ObjectAttr</a> ]	Shows the list of attributes of the object type. It will display one <code>ObjectAttr</code> class per attribute containing all the attribute values.

### GetObjectInstAttrsResult

This class returns the list of attributes of the object type instance. It displays the following information:

Parameter	Type	Description
Values	LIST[ <a href="#">ObjectInstAttrs</a> ]	The list of instances. It will display one <code>ObjectInstAttrs</code> class per instance containing all the instance values.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### GetObjectInstsResult

This class returns the list of instances of a specific object type. It displays the following information:

Parameter	Type	Description
insts	List[ <a href="#">ObjectInst</a> ]	The list of instances. It will display one <code>ObjectInst</code> class per instance containing all the instance values.

### ObjectAttr

This class returns an attribute of an object type with all its parameters.

Parameter	Type	Description
<b>Name</b>	string	The attribute name and its parameter value. Depending on the attribute there might be more than one entry.

### ObjectInst

This class returns an instance of the object type attribute. It displays the following information:

Parameter	Type	Description
id	integer	The unique BCM database identifier of the instance.

Parameter	Type	Description
<b>Name</b>	string	The name of the instance.
smallIcon	string	The name and relative path of the small icon file to be used to represent this device.

### ObjectInstAttrs

This class returns the list of attributes of an object type instance. It displays the following information:

Parameter	Type	Description
AttrName	string int boolean	The attribute name and its parameter value. Depending on the attribute/object type there might be more than one entry.

### ObjectResult

This class returns the unique object type identifier of the new object type.

Parameter	Type	Description
ObjectTypeId	integer	The unique BCM database identifier of the new object type.

### ObjectTypeAttr

This class provides the list of attributes and their values to be set for the object type instance on the device. Depending on the object type there might be more than one.

Parameter	Type	Description
AttrName	string int boolean	The name of the attribute to be modified. Depending on the object type attribute its new value can be any of the three available data types.

### PutObjectAttr

This class provides the necessary information of the attribute to be created for a specific object type.

Parameter	Type	Description
AttrName	integer	The name of the new attribute.
AttrType	integer	The data type of the new attribute.

### PutObjectAttrResult

This class lists the information about the attributes that were created for a specific object type including a list with all the new parameters and their values.

Parameter	Type	Description
ObjectTypeId	integer	The unique BCM database identifier of the object type.
AttrName	integer	The name of the new attribute.
AttrType	integer	The data type of the new attribute.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### PutObjectResult

This class returns if the creation of the new object type was successful. It displays the following information:

Parameter	Type	Description
Values	object[ <a href="#">ObjectResult</a> ]	The list of new object types together with their unique IDs.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

## Devices

The devices object type allows you to execute operations on a group of devices.

[Devices operations](#)

[Return Classes](#)

## Operations on Devices

The following operations are available on devices:

### GET

The following get operations are available:

Action	Description
<a href="#">/devices</a>	Search for devices in the BCM database.

### PUT

The following put operations are available:



Action	Description
<a href="#">/device/attr</a>	Creating a new device attribute.
<a href="#">/device/attr/{attrId}</a>	Creating a new device attribute.

## DELETE

The following delete operations are available:

Action	Description
<a href="#">/device/attr/{attrId}</a>	Creating a new device attribute.

## GET Operations

### /devices

This operation finds specific devices in the BCM database.

### Syntax

GET /devices?

source=SearchSource&query=QueryExpression&offset=OffsetStart&count=RecordCount

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
source	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The base that needs to be searched, in this case this is always <i>index</i> .

Name	Type	Mand.	Data Type	Description
query	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The complete or partial name of device(s) to be found in the database. The search is not case sensitive.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [DeviceInfo](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
16002	SQL execution error
16010	Login failed.
16401	A memory error occurred.
16402	No results were found.

### Example

This example searches all devices in whose name the letter combination "MySpy" in any combination of small or capital letters displays via the index.

#### Request URL

`https://localhost:1611/api/1/devices?source=index&query=bmc`

#### Response Body

```
{
  "1000": {
    "DeviceNetBIOSName": "MySpy-CB7L5S1",
    "DeviceIpAddress": "10.5.159.41",
    "MacAddress": "24:77:03:45:AA:18",
    "DeviceName": "MySpy-CB7L5S1",
    "DomainName": "MySpy-CB7L5S1.myspy.company.com",
    "OperatingSystemName": "Microsoft Windows 7 Enterprise Edition 64-bit",
    "User": "",
    "SerialNumber": "CB7L5S1",
    "TopologyType": "_DB_DEVTYPE_MASTER_",
    "ObjectSmallIcon": "device/master_green_win.png",
    "Type": "laptop",
    "IndexFileHit": 49,
    "IndexFileScore": 100,
    "IndexContexts": {
      "Device": "Application). Software SDLX, version 2007.03, copyright SDL International
(Application). Software Security Update for Microsoft InfoPath 2010 (KB2553322) 32-Bit Edition,
version , copyright Microsoft Corporation Inc. (Add/Remove Programs). Software <term conref =
\"../variables/variables.xml#id10B8A000W5Z/term_FullName\"/> Package Viewer, version 11.5.0.121004,
copyright BMC © Software Inc (Application). Software Device Pairing Application, version 6.1,
copyright Microsoft Corporation Inc. (Application). Software Security Update"
      "Device FAM": "sku: 2568 status: Deployed po: 41587 contact: admin vendor: Dell Filename:
Device FAM 1000 "
    }
  },
  "Total": 1,
  "Date": 1364302800,
  "Status": "_STATUS_IDLE_",
  "ErrorCode": 0
}
```

In this database there is only one device fitting the query, which is the master itself, and the search term displays in the general device data as well as within its financial information.

## Devices Return Classes

The following classes can be returned by the different devices actions:

Class	Description
<i>Device</i>	Returns the search result for one specific device.
<i>IndexContents</i>	Return an individual search result.
<i>DeviceInfo</i>	Return an individual search result.
<i>OperationResult</i>	Returns the result (error code) of the operation.

### Device (devices)

This class returns the search result of one specific device with some of its available information. Depending on the type of device and operating system not all of the below listed parameters might be included in the return class.

Parameter	Type	Description
DeviceNetBIOSName	string	The NetBIOS name of the client. For managed devices which have Linux or MAC OS as their operating system this entry will not be displayed.
DeviceIpAddress	string	The IP address of the device.
MacAddress	string	The MAC (hardware) address of the device.
DeviceName	string	The name of the device under which it is known in the BCM database.
DomainName	string	The full name of the device including the domain it belongs to, i.e., <code>kirk.enterprise.starfleet.com</code> .
OperatingSystemName	string	The name of the operating system installed on the currently selected device.
User	string	The login name of the user who is either currently connected to the device, or, if no user is connected, the user who last connected to the device.
SerialNumber	string	The serial number of the device.
TopologyType	string	The topology type of the device, that is, if the managed device is a master, a relay or a simple client. It may also be an unconnected, a scanned, a deprecated or an unknown device.

Parameter	Type	Description
ObjectSmallIcon	string	The name and relative path of the icon representing the current state of the found object.
Type	string	The type of the device, that is, a server, a laptop, a workstation, and so on.
IndexFileHit	string	Displays how often the search expression was found for the device, for example, Hits: 4, indicates that the search expression was found 4 times in different fields of the device.
IndexFileScore	string	Displays the matching percentage of the entered criteria with the search expression found on the device, for example, 100 for a full match (100%).
IndexContexts	Object[IndexContents]	Provides the search result.

### IndexContexts

This class displays the individual result for a search.

Parameter	Type	Description
Device	string	Returns the search result for one specific device.
Device FAM	string	Returns the financial search result for one specific device.

### DeviceInfo

This class returns the search result for a specific device name pattern within the BCM database.

Parameter	Type	Description
Devices	Array[Device]	Returns the list of devices found matching the search criteria.
Total	integer	The total number of devices found matching the search query.
Date	integer	The date and time at which the search was executed in its OTC format.
Status	string	The current status of the search mechanism in its keyword format.

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

## Device API

The operations of this group provide you the possibility to execute different types of actions on one specific device.

[Device Operations](#)

[Return Classes](#)

## Device Operations

The following operations are available on the device object:

### GET

The following get operations are available:

Action	Description
<a href="#">/device/{deviceId}</a>	Get attributes of device.
<a href="#">/device/{deviceId}/financial</a>	Get financial asset summary.
<a href="#">/device/{deviceId}/financial/summary</a>	Get financial asset summary.
<a href="#">/device/{deviceId}/summary</a>	Get asset summary of a device.
<a href="#">/device/{deviceId}/inventory/software/summary</a>	Get asset summary software.
<a href="#">/device/{deviceId}/inventory/catalog/summary</a>	Get asset summary catalog license.
<a href="#">/device/{deviceId}/inventory/hardware/summary</a>	Get asset summary hardware.
<a href="#">/device/{deviceId}/object/{objectId}/insts</a>	Get the inventory instances for a device and an object type.
<a href="#">/device/{deviceId}/windows/services</a>	Get services of a device.
<a href="#">/device/{deviceId}/processes</a>	Get processes of a device.
<a href="#">/device/{deviceId}/windows/registry/keys</a>	Get registry keys.
<a href="#">/device/{deviceId}/windows/registry/values</a>	Get registry values.
<a href="#">/device/{deviceId}/windows/event/categories</a>	Get windows event categories for a device.
<a href="#">/device/{deviceId}/windows/events</a>	Get windows events for a device.

Action	Description
<a href="#">/device/{deviceId}/filteredactions</a>	Return the list of actions and their availability, depending on the device properties and the administrator capabilities.
<a href="#">/device/{deviceId}/oprule/history</a>	Return the list of operational rule assigned to a device.
<a href="#">/device/{deviceId}/jnlp</a>	Create a JNLP for launching the console.
<a href="#">/device/{deviceId}/connectivity</a>	Check connectivity of a device.
<a href="#">/device/{deviceId}/fs</a>	Returns the list of drives of a device or the files and folders of a folder.
<a href="#">/device/{deviceId}/fs/file</a>	Returns content of specific file.

## POST

The following post operations are available:

Action	Description
<a href="#">/device/{deviceId}/windows/service</a>	Manage services of a device.
<a href="#">/device/{deviceId}/ping</a>	Ping a device.
<a href="#">/device/{deviceId}/shutdown</a>	Shut down a device or reboot it.
<a href="#">/device/{deviceId}/wakeup</a>	Wake a device up.
<a href="#">/device/{deviceId}/audit</a>	Audit a device.

## PUT

The following put operations are available:

Action	Description
<a href="#">/device/{deviceId}/process</a>	Run process on a device.
<a href="#">/device</a>	Create a new device.
<a href="#">/device/{deviceId}</a>	Get attributes of device.
<a href="#">/device/{deviceId}/financial</a>	Set financial attributes of device.
<a href="#">/device/{deviceId}/windows/registry/key</a>	Create Windows registry key.
<a href="#">/device/{deviceId}/windows/registry/value</a>	Create Windows registry values.
<a href="#">/device/{deviceId}/inventory/patch/scan</a>	Launch a patch scan on a device.



Action	Description
<i>/device/{deviceId}/object/{objectId}/inst</i>	Add an instance to an inventory object type for a specific device.
<i>/device/{deviceId}/object/{objectId}/inst/{instId}/attrs</i>	Specify attributes for an inventory object type instance.
<i>/device/{deviceId}/oprule/{opruleId}</i>	Assign operational rule to device.
<i>/device/{deviceId}/session</i>	Open a session on a device.
<i>/device/{deviceId}/fs/dir</i>	Create/Rename directory at the specified path.
<i>/device/{deviceId}/fs/file</i>	Copy/Rename a file in the specified path.

## DELETE

The following delete operations are available:

Action	Description
<i>/device/{deviceId}/process/{processIds}</i>	Terminate one or more processes.
<i>/device/{deviceId}</i>	Delete a specific device.
<i>/device/{deviceId}/windows/registry/key</i>	Delete a specific Windows registry key.
<i>/device/{deviceId}/windows/registry/value</i>	Delete a specific Windows registry value.
<i>/device/{deviceId}/object/{objectId}/inst/{instId}</i>	Delete an inventory object type instance on a device.
<i>/device/{deviceId}/fs/dir</i>	Create/Rename directory at the specified path.
<i>/device/{deviceId}/fs/file</i>	Copy/Rename a file in the specified path.

## GET Operations

### **/device/{deviceId}**

This operation returns all attributes of a specific device.

### Syntax

```
GET /device/{deviceId}
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

## Returns

Returns the class *DeviceAttributesInfo*.

## Comments

Multiple values are not allowed.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
10	The requested object could not be found.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.

## Example

This example finds all attributes of the master device:

**Request URL**

```
https://localhost:1611/api/1/device/1000
```

**Response Body**

```
{
  "attrs": [
    {
      "name": "_DB_ATTR_OSD_PARENT_",
      "column": "OsDeployerParent",
      "type": "Integer",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 0,
      "displayInList": false,
      "needTranslation": false,
      "valueCheckRegex": null,
      "id": 10048
    },
    {
      "name": "_DB_ATTR_SMALLICON_",
      "column": "CurrentSmallIcon",
      "type": "Enum",
      "enumName": "DeviceIcons",
      "units": null,
      "readOnly": false,
      "displayOrder": 0,
      "displayInList": false,
      "needTranslation": true,
      "valueCheckRegex": null,
      "id": 10025
    },
    {
      "name": "_DB_ATTR_LARGEICON_",
      "column": "CurrentLargeIcon",
      "type": "String",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 0,
      "displayInList": false,
      "needTranslation": false,
      "valueCheckRegex": null,
      "id": 10026
    },
    {
      "name": "_DB_ATTR_ENTRYDN_",
      "column": "LdapServerEntryDN",
      "type": "String",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 0,
      "displayInList": false,
      "needTranslation": false,
      "valueCheckRegex": null,
      "id": 10029
    },
    {
      "name": "_DB_ATTR_DEV_GUID_",
      "column": "GloballyUniqueID",
      "type": "String",
      "enumName": null,
      "units": null,
      "readOnly": false,
```

```

    "displayOrder": 0,
    "displayInList": false,
    "needTranslation": false,
    "valueCheckRegex": null,
    "id": 10030
  },
  {
    "name": "_DB_ATTR_DEV_RETIREDBY_",
    "column": "RetiredAdmin",
    "type": "String",
    "enumName": null,
    "units": null,
    "readOnly": false,
    "displayOrder": 0,
    "displayInList": false,
    "needTranslation": false,
    "valueCheckRegex": null,
    "id": 10038
  },
  {
    "name": "_DB_ATTR_DEVICE_FAMASSETTAG_",
    "column": "DeviceID",
    "type": "String",
    "enumName": null,
    "units": null,
    "readOnly": false,
    "displayOrder": 0,
    "displayInList": false,
    "needTranslation": false,
    "valueCheckRegex": "AssetTag",
    "id": 10052
  },
  {
    "name": "_DB_ATTR_DEVICE_PRIMARYUSER_",
    "column": "DeviceID",
    "type": "String",
    "enumName": null,
    "units": null,
    "readOnly": false,
    "displayOrder": 0,
    "displayInList": false,
    "needTranslation": false,
    "valueCheckRegex": "UserName",
    "id": 10053
  },
  {
    "name": "_DB_ATTR_NAME_",
    "column": "DeviceName",
    "type": "String",
    "enumName": null,
    "units": null,
    "readOnly": false,
    "displayOrder": 1,
    "displayInList": true,
    "needTranslation": false,
    "valueCheckRegex": null,
    "id": 10001
  },
  {
    "name": "UserName",
    "column": "UserName",
    "type": "String",
    "enumName": null,
    "units": null,
    "readOnly": false,
    "displayOrder": 2,
    "displayInList": true,
    "needTranslation": false,
    "valueCheckRegex": null,
    "id": 10040
  }

```

```

    },
    {
      "name": "FAMLocation",
      "column": "DeviceID",
      "type": "String",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 3,
      "displayInList": true,
      "needTranslation": false,
      "valueCheckRegex": "FAMLocation",
      "id": 10051
    },
    {
      "name": "_DB_ATTR_DEV_UPDATETIME_",
      "column": "LastUpdate",
      "type": "Date",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 4,
      "displayInList": false,
      "needTranslation": false,
      "valueCheckRegex": null,
      "id": 10019
    },
    {
      "name": "_DB_ATTR_DEV_TYPE_",
      "column": "DeviceType",
      "type": "Enum",
      "enumName": "DeviceType",
      "units": null,
      "readOnly": false,
      "displayOrder": 5,
      "displayInList": true,
      "needTranslation": true,
      "valueCheckRegex": null,
      "id": 10044
    },
    {
      "name": "_DB_ATTR_DEVICE_IPADDRESS_",
      "column": "IPAddress",
      "type": "String",
      "enumName": null,
      "units": null,
      "readOnly": false,
      "displayOrder": 6,
      "displayInList": true,
      "needTranslation": false,
      "valueCheckRegex": null,
      "id": 10005
    },
    {
      "name": "_DB_ATTR_DEVICE_TOPOLOGYTYPE_",
      "column": "TopologyType",
      "type": "Enum",
      "enumName": "DeviceTopologyTypes",
      "units": null,
      "readOnly": false,
      "displayOrder": 7,
      "displayInList": true,
      "needTranslation": true,
      "valueCheckRegex": null,
      "id": 10009
    },
    {
      "name": "_DB_ATTR_DEV_DOMAINNAME_",
      "column": "DomainName",
      "type": "String",

```

```

        "enumName": null,
        "units": null,
        "readOnly": false,
        "displayOrder": 8,
        "displayInList": true,
        "needTranslation": false,
        "valueCheckRegex": null,
        "id": 10028
    },
    { ... many more attributes ...
    {
        "name": "_DB_ATTR_NOTES_",
        "column": "Notes",
        "type": "Text",
        "enumName": null,
        "units": null,
        "readOnly": false,
        "displayOrder": 45,
        "displayInList": false,
        "needTranslation": false,
        "valueCheckRegex": null,
        "id": 10024
    }
    ]
}

```

### **/device/{deviceId}/financial**

This operation finds all financial attributes and their values of a specific device.

### **Syntax**

GET /device/{deviceId}/financial

### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

## Returns

Returns the class *DeviceFinancialAttributes*.

## Comments

Multiple values are not allowed.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
16010	Login failed.

## Example

This example finds all financial attributes of the master device:

### Request URL

```
https://localhost:1611/api/1/device/1000/financial
```

### Response Body

```
{
  "FAMID": 1000,
  "LifeCycleStatus": "Deployed",
  "AssetUser": "",
  "AssetAdmin": "",
  "Department": "",
  "Vendor": "",
  "VendorSKU": "",
  "PONumber": "",
  "WarrantyExpirationDate": "",
  "WarrantyCost": 0,
  "SupportCost": 0,
  "SupportProvider": "",
  "SupportProviderPhoneNumber": "",
  "PurchaseDate": "2013/02/22 10:02:03",
  "DateReceived": "2013/02/22 10:02:03",
  "ServiceStartDate": "2013/02/22 10:02:03",
  "PurchaseCostofAsset": 0,
  "ResidualValue": 0,
  "UsefulLife": 36,
  "LeaseCostofAsset": 0,
  "LeaseTerm": 0,
  "IsPurchased": true,
  "DepreciationType": "_STRAIGHTLINE_",
  "SupportExpirationDate": "",
  "FAMLocation": "",
  "AssetTag": "",
  "InvoiceNumber": "",
  "InvoiceDate": "",
  "ErrorCode": 0
}
```

## /device/{deviceId}/financial/summary

This operation provides a overview over the financial assets of a specific device.

### Syntax

```
GET /device/{deviceId}/financial/summary?
language=LanguageChoice&fplogin=ExtIntAdminLogin
```



## Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class [DeviceAssetFinancialSummary](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.
16240	Device not found

## Example

This example provides the financial summary of the master device:

### Request URL

```
https://localhost:1611/api/1/device/1000/financial/summary?language=English
```

### Response Body

```
{
  "PurchaseCostofAsset": 0,
  "ServiceStartDate": "2013-02-22T10:02:03Z",
  "SupportProvider": "",
  "SupportProviderPhoneNumber": "",
  "SupportExpirationDate": "",
  "WarrantyExpirationDate": "",
  "ResidualValue": 0,
  "ActualCost": 0,
  "KeywordTranslation": {
    "PurchaseCostofAsset": "Purchase Cost of Asset",
    "ServiceStartDate": "Service Start Date",
    "SupportProvider": "Support Provider",
    "SupportProviderPhoneNumber": "Support Provider Phone Number",
    "SupportExpirationDate": "Support Expiration Date",
    "WarrantyExpirationDate": "Warranty Expiration Date",
    "ResidualValue": "Residual Value",
    "ActualCost": "Current Estimated Value"
  },
  "ErrorCode": 0
}
```

## /device/{deviceId}/summary

This operation provides an overview over the main attributes of a specific device.

### Syntax

GET /device/{deviceId}/summary?language=LanguageChoice&fplogin=ExtIntAdminLogin

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

### Returns

Returns the class *DeviceAssetSummaryResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.
16240	Device not found

## Example

This example provides a summary view of the main attributes of the master device:

### Request URL

```
https://localhost:1611/api/1/device/1000/summary?language=English
```

### Response Body

```
{
  "DEVICEID": "1000",
  "DEVICENAME": "BMC-CB7L5S1",
  "IPADDRESS": "10.5.159.27",
  "MACADDRESS": "24:77:03:45:AA:18",
  "TOPOLOGYTYPE": "_DB_DEVTYPE_MASTER_",
  "VIMMACHINEVENDOR": "NotVirtual",
  "HYPERVISORVENDOR": "NotHypervisor",
  "MANUFACTURER": "Dell Inc.",
  "MODEL": "Precision M4600",
  "SERIALNUMBER": "CB7L5S1",
  "LASTUPDATE": "2013-04-29T09:30:23Z",
  "TOTALDISK": "244191",
  "LOGICALDISKSSPACE": "D:=122100 MB, C:=122094 MB",
  "LOGICALDISKSFREESPACE": "D:=9449 MB, C:=55211 MB",
  "PROCESSOR": "Intel(R) Core(TM) i7-2820QM CPU @ 2.30GHz",
  "VIDEOCONTROLLER": "NVIDIA Quadro 1000M ",
  "TOTALMEMORY": "8072",
  "NBOFPROCESSORS": "1",
  "PHYSICALMEMORY": "BANK 2- 4096 MB-64 bits, BANK 0- 4096 MB-64 bits",
  "NETWORKADAPTER": "Intel(R) Centrino(R) Ultimate-N 6300",
  "AGN/24:77:03:45:AA:18/10.5.159.27,fe80::80b0:5a58:e590:502f",
  "DOMAINNAME": "BMC-CB7L5S1.adprod.bmc.com",
  "OPERATINGSYSTEMNAME": "Microsoft Windows 7 Enterprise Edition 64-bit",
  "OPERATINGSYSTEMVERSION": "6.1.7600",
  "LASTLOGGEDINUSER": "",
  "ErrorCode": 0
}
```

## /device/{deviceId}/inventory/software/summary

This operation finds all software installed on a specific device. You can limit the list to a specific type of software, such as those installed by the Add/Remove Program functionality or Web Browsers for example.

### Syntax

GET /device/{deviceId}/inventory/software/summary?offset=OffsetNumber&count=EntryCount&language=LanguageChoice&filterparam=SoftwareFilter&fplogin=ExtIntAdminLogin

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.

Name	Type	Mand.	Data Type	Description
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.
filterparam	query	no	string	Select the type of software to display or via which way it was installed from the list. You can leave the field empty to display the result for all types.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class *DeviceAssetSoftwareSummary*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.
16240	Device not found

## Example

This example finds all installed software programs of type Web Browser, the list is limited to the first 10 and the keywords, providing the information for the respective software are translated to Spanish after the software list.

### Request URL

```
https://localhost:1611/api/1/device/1000/inventory/software/summary?offset=0&count=10&language=English&filterparam=Web+Browser
```

### Response Body

```
{
  "_DB_OBJECTTYPCLASS_SOFTWAREINVENTORY_": [
    {
      "Name": "Microsoft Internet Explorer",
      "Version": "5.10",
      "Software Manufacturer": "Skype Technologies S.A.",
      "Type": "Web Browser",
      "Installation Directory": "C:\\Program Files (x86)\\Skype\\Toolbars\\Internet Explorer"
    },
    {
      "Name": "Microsoft Internet Explorer",
      "Version": "8.0",
      "Software Manufacturer": "Microsoft Corporation Inc.",
      "Type": "Web Browser",
      "Installation Directory": "C:\\Program Files (x86)\\Internet Explorer"
    },
    {
      "Name": "Mozilla Firefox",
      "Version": "",
      "Software Manufacturer": "",
      "Type": "Web Browser",
      "Installation Directory": "C:\\Program Files (x86)\\Mozilla Firefox"
    },
    {
      "Name": "Mozilla Firefox",
      "Version": "17.0",
      "Software Manufacturer": "Mozilla Corporation",
      "Type": "Web Browser",
      "Installation Directory": "C:\\Program Files (x86)\\Mozilla Firefox"
    }
  ],
  "Total": 4,
  "KeywordTranslation": {
    "Name": "Nombre",
    "Version": "Versión",
    "Software Manufacturer": "Fabricante del software",
    "Type": "Tipo",
    "Installation Directory": "Directorio de instalación"
  },
  "ErrorCode": 0
}
```

### **/device/{deviceId}/inventory/catalog/summary**

This operation finds all software that is installed on a specific device and that is part of the software catalog.

#### **Syntax**

```
GET /device/{deviceId}/inventory/catalog/summary?
offset=OffsetNumber&count=SoftwareCount&
language=LanguageChoice&fplogin=ExtIntAdminLogin
```

#### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	ID of the device to fetch the list of all installed software that is part of the software catalog.
offset	query	no	integer	The offset from where on to start displaying the software entries. Leave this field empty together with the count field to display all entries.
count	query	no	integer	The number of software entries to display at a time. Leave this field empty together with the offset field to display all entries.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.



## Returns

Returns the class *DeviceAssetCatalogSummary*.

## Comments

This action supports pagination, by passing an offset and a number of returned values (count).

## Possible Error Codes

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.
16240	Device not found

## Example

This example requests the software catalog of the master device, with no other than the default parameters specified and the number of installed software shown limited to 10.

### Request URL

```
https://localhost:1611/api/1/device/1000/inventory/catalog/summary?count=10&language=English
```

### Response Body

```
{
  "_DB_OBJECTTYPCLASS_SOFTWAREINVENTORY_": [
    {
      "Name": "Adobe Acrobat",
      "Version": "X",
      "Software Manufacturer": "Adobe Systems Incorporated",
      "Type": "Application",
      "Category": "Desktop publishing"
    },
    {
      "Name": "Adobe Air",
      "Version": "2.x/3.x",
      "Software Manufacturer": "Adobe Systems Incorporated",
      "Type": "Application",
      "Category": "Development environment"
    },
    {
      "Name": "Adobe ExtendScript Toolkit",
      "Version": "CS5.5",
      "Software Manufacturer": "Adobe Systems Incorporated",
      "Type": "Application",
    }
  ]
}
```

```

    "Category": "Web platform development"
  },
  {
    "Name": "Adobe FrameMaker",
    "Version": "10.0",
    "Software Manufacturer": "Adobe Systems Incorporated",
    "Type": "Application",
    "Category": "Desktop publishing"
  },
  {
    "Name": "Adobe Help",
    "Version": "3.5",
    "Software Manufacturer": "Adobe Systems Incorporated",
    "Type": "Application",
    "Category": "Device drivers or system"
  },
  {
    "Name": "Adobe LiveCycle Designer ES2",
    "Version": "9.0",
    "Software Manufacturer": "Adobe Systems Incorporated",
    "Type": "Application",
    "Category": "Desktop publishing"
  },
  {
    "Name": "Adobe Reader",
    "Version": "9.3/9.4",
    "Software Manufacturer": "Adobe Systems Incorporated",
    "Type": "Application",
    "Category": "Desktop publishing"
  },
  {
    "Name": "Adobe Reader",
    "Version": "X",
    "Software Manufacturer": "Adobe Systems Incorporated",
    "Type": "Application",
    "Category": "Desktop publishing"
  },
  {
    "Name": "Akamai Download Manager",
    "Version": "2.2",
    "Software Manufacturer": "Akamai Technologies",
    "Type": "Application",
    "Category": "Network monitoring"
  },
  {
    "Name": "Alps Pointing-device Driver",
    "Version": "7.3",
    "Software Manufacturer": "Alps Electric Co. Ltd.",
    "Type": "Application",
    "Category": "Device drivers or system"
  }
],
"KeywordTranslation": {
  "Name": "Name",
  "Version": "Version",
  "Software Manufacturer": "Software Manufacturer",
  "Type": "Type",
  "Category": "Category Name"
},
"Total": 77,
"ErrorCode": 0
}

```

The response body shows the list of the first 10 installed software products found together with some basic information as well as at the bottom the total number of software products that were found installed and in the catalog, in this example 77 of them.

**/device/{deviceId}/inventory/hardware/summary**

This operation finds all hardware of a specific device.

**Syntax**

GET /device/{deviceId}/inventory/hardware/summary?language=LanguageChoice&fplogin=ExtIntAdminLogin

**Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.

Name	Type	Mand.	Data Type	Description
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class *DeviceAssetHardwareSummary*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.
16240	Device not found

## Example

This example finds all installed software programs of type Web Browser, the list is limited to the first 10 and the keywords, providing the information for the respective software are translated to Spanish after the software list.

### Request URL

```
https://localhost:1611/api/1/device/1000/inventory/hardware/summary?language=Deutsch
```

### Response Body

```
{
  "BIOS": [
    {
      "Description": "Default System BIOS"
    }
  ],
  "Cache Memory": [
    {
      "InstalledSize": "8192 KB"
    },
    {
      "InstalledSize": "256 KB"
    },
    {
      "InstalledSize": "32 KB"
    }
  ],
  "CDROM Drive": [
```

```

    {
      "Drive": "E:"
    }
  ],
  "Computer System": [
    {
      "Domain": "adprod.bmc.com",
      "Name": "BMC-CB7L5S1"
    }
  ],
  "Computer System Product": [
    {
      "Name": "Precision M4600"
    }
  ],
  "Desktop Monitor": [
    {
      "Name": "Default Monitor",
      "ScreenHeight": "0",
      "ScreenWidth": "0"
    },
    {
      "Name": "Default Monitor",
      "ScreenHeight": "0",
      "ScreenWidth": "0"
    },
    {
      "Name": "Default Monitor",
      "ScreenHeight": "1080",
      "ScreenWidth": "1920"
    },
    {
      "Name": "Generic PnP Monitor",
      "ScreenHeight": "0",
      "ScreenWidth": "0"
    }
  ],
  "Disk Drive": [
    {
      "Name": "\\.\.\.\.\PHYSICALDRIVE0",
      "Size": "244191 MB"
    }
  ],
  "Disk Partition": [
    {
      "Name": "Disk #0, Partition #1",
      "Size": "122101 MB"
    },
    {
      "Name": "Disk #0, Partition #0",
      "Size": "122095 MB"
    }
  ],
  "Keyboard": [
    {
      "Name": "Enhanced (101- or 102-key)"
    }
  ],
  .... many more attributes ....

  "Device": {
    "DeviceIPAddress": "10.5.159.27",
    "MacAddress": "24:77:03:45:AA:18",
    "SerialNumber": "CB7L5S1",
    "MostUsedUser": "",
    "Type": "Laptop",
    "DeviceName": "BMC-CB7L5S1",
    "LastUpdate": "2013-04-29T09:55:25Z",
    "OperatingSystemName": "Microsoft Windows 7 Enterprise Edition 64-bit 6.1",

```

```

"TopologyType": "_DB_DEVTYPE_MASTER_",
"DomainName": "BMC-CB7L5S1.adprod.bmc.com",
"_DB_ATTR_VIMVENDOR_": "Physisches Gerät",
"_DB_ATTR_HYPERVISORVENDOR_": "Kein Hypervisor",
"Domain": "adprod.bmc.com",
"Model": "Precision M4600",
"AssetAdmin": "",
"AssetUser": ""
},
"KeywordTranslation": {
  "Description": "Beschreibung",
  "InstalledSize": "Installierte Größe",
  "Drive": "Laufwerk",
  "Domain": "Domäne",
  "Name": "Name",
  "ScreenHeight": "Bildschirmhöhe",
  "ScreenWidth": "Bildschirmbreite",
  "Size": "Größe",
  "PrimaryBusType": "Hauptbustyp",
  "IPAddress": "IP-Adresse",
  "MACAddress": "MAC-Adresse",
  "Caption": "Beschriftung",
  "CSDVersion": "Details der Betriebssystemversion",
  "SystemUpTime": "Systemlaufzeit (Sek)",
  "BankLabel": "Steckplatzetikett",
  "Capacity": "Kapazität",
  "MaxCapacity": "Maximalkapazität",
  "CurrentClockSpeed": "Aktuelle Taktfrequenz",
  "AdapterRAM": "Adapter-RAM",
  "BIOS": "BIOS",
  "Cache Memory": "Cache-Speicher",
  "CDROM Drive": "CD-ROM Laufwerk",
  "Computer System": "Computersystem",
  "Computer System Product": "Computerprodukt",
  "Desktop Monitor": "Bildschirm",
  "Disk Drive": "Festplattenlaufwerk",
  "Disk Partition": "Festplattenpartition",
  "Keyboard": "Tastatur",
  "Logical Disk": "Logische Festplatte",
  "Motherboard Device": "Hauptplatine",
  "Network Adapter Configuration": "Netzwerkadapterkonfiguration",
  "Operating System": "Betriebssystem",
  "OS Performance": "Betriebssystemleistung",
  "Parallel Port": "Paralleler Port",
  "Physical Memory": "Physischer Speicher",
  "Physical Memory Array": "Realer Speicherarray",
  "Pointing Device": "Zeigegerät",
  "Processor": "Prozessor",
  "Sound Device": "Audiogerät",
  "Video Controller": "Videocontroller",
  "Network Adapter": "Netzwerkadapter",
  "DeviceIPAddress": "IP-Adresse",
  "MacAddress": "MAC-Adresse",
  "SerialNumber": "Seriennummer",
  "MostUsedUser": "Gerätebenutzer",
  "Type": "Typ",
  "DeviceName": "Gerätename",
  "LastUpdate": "Letzte Aktualisierung",
  "Model": "Modell",
  "OperatingSystemName": "Betriebssystem",
  "TopologyType": "Topologietyp",
  "Device": "Gerät",
  "DomainName": "Domänenname",
  "_DB_ATTR_VIMVENDOR_": "Virtualisiert auf",
  "_DB_ATTR_HYPERVISORVENDOR_": "Hosted einen Hypervisor",
  "AssetAdmin": "Verwaltet von",
  "AssetUser": "Gerätebesitzer"
},
"ErrorCode": 0
}

```

**/device/{deviceId}/object/{objectId}/insts**

This operation finds all instances of a specific object type for a given device.

**Syntax**

GET /device/{deviceId}/object/{objectId}/insts

**Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type. In the appendix you can find a list with the most commonly used object type IDs. Other object ids are returned via the <code>GET /objects</code> operation.

## Returns

Returns the class *DeviceInventoryInsts*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error



Error Code	Description
16010	Login failed.
16040	Bad object type.

## Comments

Multiple values are not allowed.

## Example

This example displays the instances of the processor (ID: 155) of the master device (ID 1000). In the following example the device only has one processor.

### Request URL

```
https://localhost:1611/api/1/device/1000/object/155/insts
```

### Response Body

```
{
  "Object": [
    {
      "ObjectId": 1000,
      "ObjectName": "0",
      "InstanceName": "0",
      "ConfigManagerErrorCode": "This device is working properly",
      "CurrentClockSpeed": "2301",
      "Description": "Intel64 Family 6 Model 42 Stepping 7",
      "ExtClock": "100",
      "Family": "198",
      "L2CacheSize": "256",
      "L2CacheSpeed": "0",
      "LoadPercentage": "",
      "Manufacturer": "Intel Corporation Inc.",
      "MaxClockSpeed": "2301",
      "Name": "Intel(R) Core(TM) i7-2820QM CPU @ 2.30GHz",
      "PowerManagementSupported": "FALSE",
      "ProcessorId": "BFEBFBFF000206A7",
      "ProcessorType": "Central Processor",
      "Status": "OK",
      "Stepping": "",
      "UniqueId": "",
      "UpgradeMethod": "Other",
      "Version": ""
    }
  ],
  "ErrorCode": 0
}
```

## /device/{deviceId}/windows/services

This operation returns the list of all services of a device.

## Syntax

```
GET /device/{deviceId}/windows/services?offset=StartEntry&count=EntryCount&
sort=SortAttribute&asc=SortOrder&lan=LanguageChoice
```

Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.
sort	query	no	string	Select from this drop-down box the attribute according to which the result is to be sorted.
asc	query	no	boolean	Select in this drop-down box if the sort order is to be ascending or descending.
lan	query	no	string	Select from this drop-down box the language in which the attributes are to be provided.

## Returns

Returns the class *DeviceServiceInfo*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## /device/{deviceId}/processes

This operation returns the list of processes running on a specific device.

## Syntax

GET /device/{deviceId}/processes?Session-Id=SessionId

## Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class [DeviceProcesses](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## Example

This example lists the processes that are currently running on the master device.

### Request URL

```
https://localhost:1611/api/1/device/1000/processes
```

### Response Body

```
{
  "Total": 107,
  "Values": [
    {
      "ProcessOwner": "",
      "ProcessName": "32bitProxy.exe",
      "ProcessPath": "",
      "ProcessTotalCpu": "",
      "ProcessCurrentCpu": 0,
      "ProcessID": "3572",
      "ProcessParentID": "1324",
      "ProcessMemoryVirtual": 0,
      "ProcessMemoryUsed": 0,
      "ProcessThreadsCount": 4
    },
    {
      "ProcessOwner": "ADPROD\\sertle",
      "ProcessName": "32bitProxy.exe",
      "ProcessPath": "D:\\Program Files\\BMC Software\\Panoramix\\bin\\opswat\\32bitProxy.exe",
      "ProcessTotalCpu": " 00:00:06",
      "ProcessCurrentCpu": 0,
      "ProcessID": "7176",
      "ProcessParentID": "7224",
      "ProcessMemoryVirtual": 53628,
      "ProcessMemoryUsed": 48320,
      "ProcessThreadsCount": 4
    },
    ...many more processes ....
  ],
  "ErrorCode": 0
}
```

**/device/{deviceId}/windows/registry/keys**

This operation returns the list of registry keys under a specific key.

**Syntax**

GET /device/{deviceId}/windows/registry/keys?path=KeyPath

**Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.



Name	Type	Mand.	Data Type	Description
path	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The path of the registry key.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class [DeviceRegistryKeys](#).

## Comments

Use only single slashes (/) in the registry path.

Depending on the Windows operating system version accessing registry root keys like HKEY\_CLASSES\_ROOT, HKEY\_CURRENT\_USER, HKEY\_LOCAL\_MACHINE, HKEY\_USERS, HKEY\_CURRENT\_CONFIG HKLM might require further administrator privileges.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## Example

This example lists all subkeys that can be found under the HKEY\_LOCAL\_MACHINE\SOFTWARE key.

### Request URL

```
https://localhost:1611/api/1/device/1000/windows/registry/keys?path=HKEY_LOCAL_MACHINE%5CSOFTWARE
```

### Response Body

```
{
  "keys": [
    {
      "name": "ATI Technologies"
    },
    {
      "name": "Adobe"
    },
    {
      "name": "Alps"
    },
    {
      "name": "BMC Software"
    },
    {
      "name": "Classes"
    },
    {
      "name": "Clients"
    },
    {
      "name": "Google"
    },
    ..... many more keys
    {
      "name": "WinRAR"
    },
    {
      "name": "Windows"
    },
    {
      "name": "Wow6432Node"
    }
  ]
}
```

```
]
}
```

**/device/{deviceId}/windows/registry/values**

This operation returns the values for a specific registry key.

**Syntax**

GET /device/{deviceId}/windows/registry/values?path=KeyPath

**Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.

Name	Type	Mand.	Data Type	Description
path	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The path of the registry key.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class [DeviceRegistryValues](#) .

## Comments

Use only single slashes (/) in the registry path.

Depending on the Windows operating system version accessing registry root keys like HKEY\_CLASSES\_ROOT, HKEY\_CURRENT\_USER, HKEY\_LOCAL\_MACHINE, HKEY\_USERS, HKEY\_CURRENT\_CONFIG HKLM might require further administrator privileges.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

### Example

This example lists all key values for the HKEY\_LOCAL\_MACHINE\SOFTWARE\BMC Software key.

#### Request URL

```
https://localhost:1611/api/1/device/1000/windows/registry/values?path=HKEY_LOCAL_MACHINE%5CSOFTWARE%5CBMC+Software
```

#### Response Body

```
{
  "values": [
    {
      "Name": "",
      "Type": "REG_SZ",
      "Data": ""
    }
  ]
}
```

### /device/{deviceId}/windows/event/categories

This operation returns the list of Windows event categories for a device.

#### Syntax

```
GET /device/{deviceId}/windows/event/categories
```

#### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.



Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class [DeviceCategoryResultValues](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## Example

This example lists all Windows Event Categories found on the master device.

### Request URL

```
https://localhost:1611/api/1/device/1000/windows/event/categories
```

### Response Body

```
{
  "DACurrentConsoleUser": "admin",
  "Values": [
    {
      "NtEventCategoryName": "Application",
      "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
      "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_APPDESC_",
      "NtEventCategorySize": -1
    },
    {
      "NtEventCategoryName": "Cisco AnyConnect VPN Client",
      "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
      "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
      "NtEventCategorySize": -1
    },
    {
      "NtEventCategoryName": "HardwareEvents",
      "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
      "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
      "NtEventCategorySize": -1
    },
    {
      "NtEventCategoryName": "Internet Explorer",
      "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
      "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
      "NtEventCategorySize": -1
    },
    {
      "NtEventCategoryName": "Key Management Service",
      "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
      "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
      "NtEventCategorySize": -1
    },
    {
      "NtEventCategoryName": "Media Center",
      "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
      "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
      "NtEventCategorySize": -1
    }
  ]
}
```

```

    "NtEventCategoryName": "OAlerts",
    "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
    "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
    "NtEventCategorySize": -1
  },
  {
    "NtEventCategoryName": "",
    "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
    "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
    "NtEventCategorySize": -1
  },
  {
    "NtEventCategoryName": "System",
    "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
    "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_SYSDDESC_",
    "NtEventCategorySize": -1
  },
  {
    "NtEventCategoryName": "Windows PowerShell",
    "NtEventCategoryType": "_MISC_WINDOWSEVENT_TYPE_",
    "NtEventCategoryDesc": "_MISC_WINDOWSEVENT_CUSTOMDESC_",
    "NtEventCategorySize": -1
  }
],
"ErrorCode": 0
}

```

### **/device/{deviceId}/windows/events**

This operation returns the list of Windows events for a specific device.

### **Syntax**

GET /device/{deviceId}/windows/events?

categoryindex=PositionNumber&offset=StartEntry&count=EntryCount

### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
categoryindex	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The index, that is, the position at which the respective category was listed via the result of the <code>GET /device/{deviceId}/windows/event/categories</code> operation. The position numbering starts at 0.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class [DeviceWindowsEventResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## Example

This example lists the first ten Application events (the first category in the list, that is, at position 0) of a total of 48686 logged events for the admin user on the master device.

### Request URL

```
https://localhost:1611/api/1/device/1000/windows/events?categoryindex=0&offset=0&count=10
```

### Response Body

```
{
  "Offset": 0,
  "Count": 10,
  "NtEventIndex": 0,
  "DACurrentConsoleUser": "admin",
  "Total": 48686,
  "Values": [
    {
      "NtEventType": "_MISC_EVENTLOG_ERROR_",
      "NtEventDateTime": "2012/05/25 10:47:18",
      "NtEventSource": "Communicator",
      "NtEventCategory": 0,
      "NtEventID": 3,
      "NtEventUser": "N/A",
      "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the login
server sip.bmc.com.\r\n\r\n\r\n\r\nResolution:\r\n\r\nIf you are using manual configuration for
Communicator, please check that the server name is typed correctly and in full. If you are using
automatic configuration, the network administrator will need to double-check the DNS A record
configuration for sip.bmc.com because it could not be resolved.\r\n"
    },
    {
      "NtEventType": "_MISC_EVENTLOG_ERROR_",
      "NtEventDateTime": "2012/05/25 10:47:22",
      "NtEventSource": "Communicator",
      "NtEventCategory": 0,
      "NtEventID": 3,
      "NtEventUser": "N/A",
      "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the login
server sip.bmc.com.\r\n\r\n\r\n\r\nResolution:\r\n\r\nIf you are using manual configuration for
Communicator, please check that the server name is typed correctly and in full. If you are using
automatic configuration, the network administrator will need to double-check the DNS A record
configuration for sip.bmc.com because it could not be resolved.\r\n"
    },
    {
      "NtEventType": "_MISC_EVENTLOG_ERROR_",
      "NtEventDateTime": "2012/05/25 10:47:25",
      "NtEventSource": "Communicator",
      "NtEventCategory": 0,
      "NtEventID": 3,

```

```

    "NtEventUser": "N/A",
    "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the
login server sipexternal.bmc.com.\r\n\r\n\r\n\r\nResolution:\r\n\r\n\r\nIf you are using manual
configuration for Communicator, please check that the server name is typed correctly and in full.
If you are using automatic configuration, the network administrator will need to double-check
the DNS A record configuration for sipexternal.bmc.com because it could not be resolved.\r\n"
  },
  {
    "NtEventType": "_MISC_EVENTLOG_ERROR_",
    "NtEventDateTime": "2012/05/25 10:47:25",
    "NtEventSource": "Communicator",
    "NtEventCategory": 0,
    "NtEventID": 3,
    "NtEventUser": "N/A",
    "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the
login server sipexternal.bmc.com.\r\n\r\n\r\n\r\nResolution:\r\n\r\n\r\nIf you are using manual
configuration for Communicator, please check that the server name is typed correctly and in full.
If you are using automatic configuration, the network administrator will need to double-check
the DNS A record configuration for sipexternal.bmc.com because it could not be resolved.\r\n"
  },
  {
    "NtEventType": "_MISC_EVENTLOG_WARNING_",
    "NtEventDateTime": "2012/05/25 10:47:55",
    "NtEventSource": "Communicator",
    "NtEventCategory": 0,
    "NtEventID": 1,
    "NtEventUser": "N/A",
    "NtEventDescription": "Communicator was unable to locate the login server. No DNS
SRV records exist for domain bmc.com, so Communicator was unable to logon.\r\n\r\n\r\n\r\n\r\n
Resolution:\r\n\r\n\r\nPlease double-check the server name to ensure that it is typed correctly.
If it is correct, the network administrator will either need to use manual configuration
to specify the login server's fully-qualified domain name (FQDN), or add DNS SRV records for
the bmc.com domain in order to allow automatic client configuration. The DNS SRV records
_sipinternaltls.tcp.bmc.com, _sipinternal.tcp.bmc.com and/or _sip.tls.bmc.com may need to be
configured if automatic configuration is desired.\r\n"
  },
  {
    "NtEventType": "_MISC_EVENTLOG_ERROR_",
    "NtEventDateTime": "2012/05/25 10:47:55",
    "NtEventSource": "Communicator",
    "NtEventCategory": 0,
    "NtEventID": 3,
    "NtEventUser": "N/A",
    "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the
login server sipinternal.bmc.com.\r\n\r\n\r\n\r\n\r\nResolution:\r\n\r\n\r\nIf you are using manual
configuration for Communicator, please check that the server name is typed correctly and in full.
If you are using automatic configuration, the network administrator will need to double-check
the DNS A record configuration for sipinternal.bmc.com because it could not be resolved.\r\n"
  },
  {
    "NtEventType": "_MISC_EVENTLOG_ERROR_",
    "NtEventDateTime": "2012/05/25 10:47:55",
    "NtEventSource": "Communicator",
    "NtEventCategory": 0,
    "NtEventID": 3,
    "NtEventUser": "N/A",
    "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the
login server sipinternal.bmc.com.\r\n\r\n\r\n\r\n\r\nResolution:\r\n\r\n\r\nIf you are using manual
configuration for Communicator, please check that the server name is typed correctly and in full.
If you are using automatic configuration, the network administrator will need to double-check
the DNS A record configuration for sipinternal.bmc.com because it could not be resolved.\r\n"
  },
  {
    "NtEventType": "_MISC_EVENTLOG_ERROR_",
    "NtEventDateTime": "2012/05/25 10:47:57",
    "NtEventSource": "Communicator",
    "NtEventCategory": 0,
    "NtEventID": 3,
    "NtEventUser": "N/A",

```



```

    "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the login
server sip.bmc.com.\r\n\r\n\r\n\r\nResolution:\r\n\r\n\r\nIf you are using manual configuration for
Communicator, please check that the server name is typed correctly and in full.  If you are using
automatic configuration, the network administrator will need to double-check the DNS A record
configuration for sip.bmc.com because it could not be resolved.\r\n"
  },
  {
    "NtEventType": "_MISC_EVENTLOG_ERROR_",
    "NtEventDateTime": "2012/05/25 10:48:02",
    "NtEventSource": "Communicator",
    "NtEventCategory": 0,
    "NtEventID": 3,
    "NtEventUser": "N/A",
    "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the login
server sip.bmc.com.\r\n\r\n\r\n\r\nResolution:\r\n\r\n\r\nIf you are using manual configuration for
Communicator, please check that the server name is typed correctly and in full.  If you are using
automatic configuration, the network administrator will need to double-check the DNS A record
configuration for sip.bmc.com because it could not be resolved.\r\n"
  },
  {
    "NtEventType": "_MISC_EVENTLOG_ERROR_",
    "NtEventDateTime": "2012/05/25 10:48:04",
    "NtEventSource": "Communicator",
    "NtEventCategory": 0,
    "NtEventID": 3,
    "NtEventUser": "N/A",
    "NtEventDescription": "Communicator was unable to resolve the DNS hostname of the
login server sipexternal.bmc.com.\r\n\r\n\r\n\r\nResolution:\r\n\r\n\r\nIf you are using manual
configuration for Communicator, please check that the server name is typed correctly and in full.
If you are using automatic configuration, the network administrator will need to double-check
the DNS A record configuration for sipexternal.bmc.com because it could not be resolved.\r\n"
  }
],
"ErrorCode": 0
}

```

### /device/{deviceId}/filteredactions

This operation return the list of actions and their availability, depending on the device properties and the administrator capabilities.

### Syntax

GET /device/{deviceId}/filteredactions?fplogin=ExtIntAdminLogin&language=LanguageChoice

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.

## Returns

Returns the class *FilteredActionsResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.
16010	Login failed.
16116	The object could not be found.
16240	Device not found

## Example

This example shows all the filtered actions available on the master device with the keywords translated to German.

### Request URL

```
https://localhost:1611/api/1/device/1000/filteredactions?language=Deutsch
```

### Response Body

```
{
  "ConfigSummary": true,
  "RemoteControl": true,
  "FileSystem": true,
  "Registry": true,
  "Events": true,
  "Processes": true,
  "Services": true,
  "PerformAudit": true,
  "FileTransfer": true,
  "Ping": false,
  "Reboot": false,
  "Shutdown": false,
  "WakeUp": false,
  "Values": {
    "ConfigSummary": "Inventarzusammenfassung",
    "RemoteControl": "Remote Control",
    "FileSystem": "Dateisystem",
    "Registry": "Registry",
    "Events": "Windows-Ereignisse",
    "Processes": "Prozeßverwaltung",
    "Services": "Dienste",
    "PerformAudit": "Jetzt prüfen",
    "FileTransfer": "Dateiübertragung",
    "Ping": "Verbindung prüfen",
    "Reboot": "Neu starten",
    "Shutdown": "Ausschalten",
    "WakeUp": "Aufwecken"
  },
  "ErrorCode": 0
}
```

## /device/{deviceId}/oprule/history

This operation returns the list of operational rule assigned to a device.

### Syntax

GET /device/{deviceId}/oprule/history?fplogin=ExtInAdminLogin&  
language=LanguageChoice&offset=Start&count=NumberOfEntries

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.



[illegible]

```
/device/{deviceId}/jnlp
```

This operation creates a JNLP file to launch the console. BMC Client Management provides a Web Service action that returns the launch JNLP file. The file contains command lines necessary to launch the BCM console and an access token. The access token is stored in memory and is an MD5 string build with the AdminId, the DeviceID, the context and the timestamp.

## Syntax

```
GET /device/{deviceId}/jnlp?ctx=Command&rctx=SCContext&fplogin=ExtIntAdminLogin
```

## Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
ctx	query	no	string	Select the name of the command on which the console is to open from the drop-down list.
rctx	query	no	string	Select the request context of the external integration if required.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class [JnlpResult](#).

## Comments

When a web service is called, the authentication is done with the *Integration Administrator* credentials. The action is executed with BMC FootPrints Service Core login administrator rights.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.

## Example

This example gets the jnlp for the Registry context.

### Request URL

<https://localhost:1611/api/1/device/1000/jnlp?ctx=Registry>

## Response Body

[illegible]

**/device/{deviceId}/connectivity**

This operation checks the connectivity of a specific device.

## Syntax

```
GET /device/{deviceId}/connectivity?validation=&fplogin=ExtIntAdminLogin
```

## Parameters

The action takes the following parameters:



Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
validation	query	no	integer	Defines if the external integration login is to be validated.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class [DeviceOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
16002	SQL execution error
16010	Login failed.
16210	The administrator account is disabled.
16211	The administrator could not be found.
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.

### Example

This example checks the connectivity of the master device.

#### Request URL

```
https://localhost:1611/api/1/device/1000/connectivity
```

#### Response Body

```
{
  "Validation": 0,
  "Version": "Version=1.0",
  "HttpProtocolRequestInfo": {
    "HttpProtocolRequestUser": "admin"
  },
  "ErrorCode": 0
}
```

### /device/{deviceId}/fs

This operation returns the list of drives of a device or the list of files and folders of a folder.

### Syntax

GET /device/{deviceId}/fs?

path=DirPath&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage&lan=SearchLanguage

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
path	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The absolute path of directory of which list of directories and files is to be returned. Use double backslashes (\\) in the path. If this field is left empty the directories and files below the root directory are returned.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the file or folder (COREUTILS_ACTPRMVALUE_COLNAME_FILE), its size (COREUTILS_ACTPRMVALUE_COLNAME_SIZE), its access (COREUTILS_ACTPRMVALUE_COLNAME_ACCESS) or the date at which it was last modified (COREUTILS_ACTPRMVALUE_COLNAME_MODIFIED).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending <i>true</i> or descending ( <i>false</i> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.

Name	Type	Mand.	Data Type	Description
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [DeviceFileSystemDirInfo](#) if the `path` parameter is provided, otherwise the class [DeviceFileSystemInfo](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## Example

This example shows the contents of the local directory D:\Program Files on the master:

### Request URL

```
https://localhost:1611/api/1/device/1000/fs?path=D%3A%5C%5CProgram
+Files&sort=COREUTILS_ACTPRMVALUE_COLNAME_FILE&asc=true&lan=english
```

### Response Body

```
{
  "path": {
    "name": "D:\\\\Program Files",
    "children": [
      {
        "name": "BMC Software",
        "isDir": true,
        "modifiedDate": null,
        "mode": 16895
      },
      {
        "name": "full_license.xml",
        "isDir": false,
        "modifiedDate": 1378819692,

```

```

    "mode": 33206,
    "size": 3111,
    "isText": true,
    "isHidden": false
  },
  {
    "name": "license.xml",
    "isDir": false,
    "modifiedDate": 1372859764,
    "mode": 33206,
    "size": 4867,
    "isText": true,
    "isHidden": false
  }
]
}
}

```

### **/device/{deviceId}/fs/file**

This operation returns the data available for a specific file.

### **Syntax**

GET /device/{deviceId}/fs/file?path=FilePath&decode=DecodeMode

### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.



Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
path	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The file name with the absolute path. Use double backslashes (\\) in the path.
decode	query	no	boolean	Select from this drop-down box if the content of the file is to be displayed in decoded, that is, legible format ( <i>true</i> ) or in its encoded version ( <i>false</i> ).

## Returns

Returns the class [DeviceReadFile](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
12	Access denied. Please verify your access rights.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.
14001	The file could not be found.

## Example

This example provides the available information about file D:\\Program Files\\license.xml, located on the master in decoded form:

### Request URL

```
https://localhost:1611/api/1/device/1000/fs/file?path=D%3A%5C%5C%5CProgram+Files%5C%5Clicense.xml&decode=true
```

### Response Body

```
{
  "path": "D:\\\\Program Files\\\\license.xml",
  "offset": 0,
  "length": 4867,
  "result": "711645B544A4DE460C0E8AA4C86F7296\\r\\n\\r\\n_DB_LICENSETYPE_TEMP_\\r\\n\\r\\n\\r\\n10.5.159.150\\r\\n\\r\\nUnknown\\r\\n11.0\\r\\n\\r\\n\\r\\n_DB_LICENSEFEATURE_DEVICECOUNT_\\r\\n30\\r\\n\\r\\n\\r\\n_DB_LICENSEFEATURE_UNMANAGED_\\r\\n30\\r\\n\\r\\n\\r\\n\\r\\n_DB_LICENSEFEATURE_INVENTORY_\\r\\n30\\r\\n\\r\\n\\r\\n_DB_LICENSEFEATURE_TOPOLOGY_\\r\\n1\\r\\n\\r\\n\\r\\n\\r\\n_DB_LICENSEFEATURE_MULTICAST_\\r\\n1\\r\\n\\r\\n\\r\\n\\r\\n_DB_LICENSEFEATURE_APPLICATIONMANAGEMENT_\\r\\n30\\r\\n\\r\\n\\r\\n\\r\\n_DB_LICENSEFEATURE_TASKMANAGEMENT_\\r\\n1\\r\\n\\r\\n\\r\\n\\r\\n\\r\\n'_DB_STEPNAME_AGENTCONFIG_', '_DB_STEPNAME_AGENTRESTART_', '_DB_STEPNAME_AGENTREBOOTPARAMETER_', '_DB_STEPNAME_'\\r\\n\\r\\n"
}
```

## POST Operations

### /device/{deviceId}/windows/service

This operation manages the services of a specific device.

### Syntax

```
POST /device/{deviceId}/windows/service?name=ServiceName&do=ActionToPerform
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
<b>Name</b>	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The name of the service, as returned by the GET /device/{deviceId}/windows/services operation.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
do	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	Select the action to perform on the specified service from the drop-down list.

## Returns

Returns the class [DeviceServiceInfo](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.
38005	Invalid service name

**/device/{deviceId}/ping**

This operation pings a device.

**Syntax**

POST /device/{deviceId}/ping

**Parameters**

The action takes the following parameters:



Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

## Returns

Returns the class [DeviceResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error

Error Code	Description
16010	Login failed.
16040	Bad object type.

**Example**

This example pings the master device with success.

**Request URL**

```
https://localhost:1611/api/1/device/1000/ping
```

**Response Body**

```
{
  "Result": 2,
  "ErrorCode": 0
}
```

**/device/{deviceId}/shutdown**

This operation shuts down or reboots a specific device.

**Syntax**

POST /device/{deviceId}/shutdown?reboot=OperationType

**Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
reboot	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The shutdown operation to execute, that is, if the device is to be rebooted (yes - default) or completely shut down (no).

## Returns

Returns the class `DeviceExit`.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

## Example

This example tries to reboot the device with BCM ID 1001 without success.

### Request URL

```
https://localhost:1611/api/1/device/1001/ping
```

### Response Body

```
{
  "fault": {
    "code": 13,
    "uri": "/api/1/device/1001/shutdown"
  }
}
```

## /device/{deviceId}/wakeup

This operation wakes up a specific device.

### Syntax

POST /device/{deviceId}/wakeup?retrycount=NumberOfRetries&retryinterval=RetryInterval

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
retrycount	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The number of retries, that is the number of times the wakeup operation is to be repeated if the wakeup fails.
retryinterval	query	no	integer	The interval in seconds between two wakeup retries.

## Returns

Returns the class [DeviceOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error

**Example**

This example wakes up device with ID 1001 with success.

**Request URL**

```
https://localhost:1611/api/1/device/1001/wakeup?retrycount=3&retryinterval=3
```

**Response Body**

```
{
  "ErrorCode": 0
}
```

**/device/{deviceId}/audit**

This operation audits a specific device.

**Syntax**

POST /device/{deviceId}/audit?rctx=SCContext&fplogin=ExtIntAdminLogin

**Parameters**

The action takes the following parameters:



Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.
rctx	query	no	string	Select the request context of the external integration if required.

## Returns

Returns the class [DeviceOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
12	Access denied. Please verify your access rights.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.
16002	SQL execution error
16010	Login failed.
33001	Device unreachable

## PUT Operations

### /device/{deviceId}/process

This operation launches a process on a device.

### Syntax

```
PUT /device/{deviceId}/process?path=C:\Program Files (x86)\Skype\Phone
\Skype.exe&parameters=Param1,Param2&context=RunContext
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
path	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The full path of the executable file to run.
parameters	body	no	string	The parameters to be passed to the executable, separated by comma(,).
context	body	no	boolean	Define if the executable is run in context.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class [DeviceOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## Example

The following example starts skype.

### Request URL

```
https://localhost:1611/api/1/device/1000/process?path=C%3A%5CProgram+Files+(x86)%5Cskype%5CPhone%5Cskype.exe
```

### Response Body

```
{
  "ProcessID": [
    "1968"
  ],
  "DACurrentConsoleUser": "admin",
  "ErrorCode": 0
}
```

## /device

This operation creates a new device in the BCM database.

## Syntax

PUT /device?name=ObjectName

The body must contain the definitions of the `DeviceData` class.

```
{ "name": "string" }
```

## Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
DeviceData	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	DeviceData	The data for the new device to be created.

## Returns

Returns the class [DeviceCreate](#).

## Comments

Be aware, that this operation only creates device but does not yet set any of its attributes. To set those you also need to execute the [PUT /device/{deviceid}](#) operation.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.
16041	Bad object name.

### Example

This example creates a new device with the name "NewTestDevice".

#### Request URL

```
https://localhost:1611/api/1/device
```

whereby the body contains the following information:

```
{
  "name": "NewTestDevice"
}
```

#### Response Body

```
{
  "ObjectId": 1001,
  "ErrorCode": 0
}
```

### /device/{deviceId}

This operation adds or modifies the attribute values of a specific device.

### Syntax

PUT /device/{deviceId}

The body must contain at least one of the definitions of the `DeviceAttributes` class.

```
{ "_DB_ATTR_OSD_PARENT_": "int", "_DB_ATTR_SMALLICON_": "string",
  "_DB_ATTR_LARGEICON_": "string", "_DB_ATTR_ENTRYDN_": "string", "_DB_ATTR_DEV_GUID_":
  "string", "_DB_ATTR_DEV_RETIREDBY_": "string", "_DB_ATTR_DEVICE_FAMASSETTAG_":
  "string", "_DB_ATTR_DEVICE_PRIMARYUSER_": "string", "_DB_ATTR_NAME_":
  "string", "UserName": "string", "_DB_ATTR_DEV_TYPE_": "string", "FAMLocation":
  "string", "_DB_ATTR_VIMVENDOR_": "string", "_DB_ATTR_HYPERVISORVENDOR_":
  "string", "_DB_ATTR_HYPERVISORVERSION_": "string", "_DB_ATTR_DEV_DOMAINNAME_":
```



```
"string", "_DB_ATTR_DEV_NETWORKNAME_": "string", "_DB_ATTR_DEV_HTTPPORT_":
"int", "_DB_ATTR_DEV_HTTPCONSOLEPORT_": "int", "_DB_ATTR_DEV_SSL_": "string",
"_DB_ATTR_DEVICE_NETBIOSNAME_": "string", "_DB_ATTR_DEVICE_IPADDRESS_":
"string", "_DB_ATTR_DEVICE_NETWORKMASK_": "string", "_DB_ATTR_DEVICE_UNDERNAT_":
"boolean", "_DB_ATTR_DEVICE_MACADDRESS_": "string", "_DB_ATTR_DEVICE_HOSTID_":
"string", "_DB_ATTR_DEVICE_DISKSERNUM_": "string", "_DB_ATTR_PARENT_":
"string", "_DB_ATTR_DEVICE_TOPOLOGYTYPE_": "string", "_DB_ATTR_DEV_OSNAME_":
"string", "_DB_ATTR_DEV_OSVERMAJOR_": "int", "_DB_ATTR_DEV_OSVERMINOR_":
"int", "_DB_ATTR_DEV_OSREVISION_": "string", "_DB_ATTR_DEV_OSBUILD_": "int",
"_DB_ATTR_DEV_AGENTVERMAJOR_": "int", "_DB_ATTR_DEV_AGENTVERMINOR_": "int",
"_DB_ATTR_DEV_AGENTREVISION_": "string", "_DB_ATTR_DEV_AGENTBUILD_": "string",
"_DB_ATTR_PATCHCONFIGFILEVERSION_": "string", "_DB_ATTR_PKGFACTORY_":
"boolean", "_DB_ATTR_PATCHPROXY_": "boolean", "_DB_ATTR_OSD_": "boolean",
"_DB_ATTR_VPROAVAILABLE_": "boolean", "_DB_ATTR_RISCANNER_": "boolean",
"_DB_ATTR_ROLLOUTSRV_": "boolean", "_DB_ATTR_DEV_UPDATETIME_": "string",
"_DB_ATTR_CREATEDBY_": "string", "_DB_ATTR_CREATETIME_": "string",
"_DB_ATTR_LASTMODIFIEDBY_": "string", "_DB_ATTR_LASTMODIFYTIME_": "string",
"_DB_ATTR_NOTES_": "string" }
```

## Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
DeviceAttributes	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	DeviceAttributes	The list of device attributes that need to be updated which can be any combination or all the parameters of the <code>DeviceAttributes</code> class.  For more information about the possible values for these parameters see <a href="#">DeviceAttributes</a> .

## Returns

Returns the class [DeviceUpdate](#).

## Comments

Multiple values are not allowed.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

Error Code	Description
16010	Login failed.
16040	Bad object type.

## Example

This example sets a number of attributes of the newly create device 1001.

### Request URL

```
https://localhost:1611/api/1/device/1001
```

whereby the body contains the following information:

```
{
  "_DB_ATTR_DEV_NETWORKNAME_": "myspy_007",
  "_DB_ATTR_DEV_HTTPPORT_": "1601",
  "_DB_ATTR_DEV_HTTPCONSOLEPORT_": "1602",
  "_DB_ATTR_PKGFACTORY_": "0",
  "_DB_ATTR_PATCHPROXY_": "1",
  "_DB_ATTR_OSD_": "0",
  "_DB_ATTR_VPROAVAILABLE_": "0",
  "_DB_ATTR_RISCANNER_": "1",
  "_DB_ATTR_ROLLOUTSRV_": "1"
}
```

### Response Body

```
{
  "ObjectTypeId": 100,
  "ObjectId": 1001,
  "ObjectAttrs": {
    "_DB_ATTR_DEV_NETWORKNAME_": "myspy_007",
    "_DB_ATTR_DEV_HTTPPORT_": "1601",
    "_DB_ATTR_DEV_HTTPCONSOLEPORT_": "1602",
    "_DB_ATTR_PKGFACTORY_": "0",
    "_DB_ATTR_PATCHPROXY_": "1",
    "_DB_ATTR_OSD_": "0",
    "_DB_ATTR_VPROAVAILABLE_": "0",
    "_DB_ATTR_RISCANNER_": "1",
    "_DB_ATTR_ROLLOUTSRV_": "1"
  },
  "ErrorCode": 0
}
```

## /device/{deviceId}/financial

This operation adds or modifies financial attributes of device.

## Syntax

```
PUT /device/{deviceId}/financial
```

The body must contain at least one of the definitions of the `DeviceFinancialAttributes` class.

```
{ "FAMID": "int", "LifecycleStatus": "string", "AssetUser": "string", "AssetAdmin":
  "string", "Department": "string", "Vendor": "string", "VendorSKU": "string",
```

```
"PONumber": "string", "WarrantyExpirationDate": "string", "WarrantyCost": "int",
"SupportCost": "int", "SupportProvider": "string", "SupportProviderPhoneNumber":
"string", "PurchaseDate": "string", "DateReceived": "string", "ServiceStartDate":
"string", "PurchaseCostofAsset": "int", "ResidualValue": "int", "UsefulLife": "int",
"LeaseCostofAsset": "int", "LeaseTerm": "int", "LeaseTermFrequency": "string",
"IsPurchased": "boolean", "DepreciationType": "string", "SupportExpirationDate":
"string", "FAMLocation": "string", "AssetTag": "string", "InvoiceNumber": "string",
"InvoiceDate": "string", "ErrorCode": "int" }
```

## Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
DeviceFinancial Attributes	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	DeviceFinancial Attributes	<p>The list of financial attributes that needs to be updated for the device. Any combination or all of the available values is possible.</p> <p>For more information about the possible values of these parameters see <a href="#">DeviceFinancialAttributes</a>.</p>

## Returns

Returns the class [DeviceFinancialAttributes](#) containing only those parameters and their new values that were updated.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
13	Incorrect parameter. Please verify the supplied parameters.

Error Code	Description
16010	Login failed.
16504	One or more statuses have not been changed.

### Example

The following example creates new financial attribute values for the new device 1001.

#### Request URL

```
https://localhost:1611/api/1/device/1001/financial
```

whereby the body contains the following information:

```
{
  "LifeCycleStatus": "Deployed",
  "Vendor": "Dell",
  "PONumber": "PO-1249852",
  "SupportProvider": "Dell",
  "IsPurchased": "1",
  "FAMLocation": "Sophia",
  "InvoiceNumber": "Dell-264895-xp"
}
```

#### Response Body

```
{
  "DeviceID": 1001,
  "LifeCycleStatus": "Deployed",
  "Vendor": "Dell",
  "PONumber": "PO-1249852",
  "SupportProvider": "Dell",
  "IsPurchased": "1",
  "FAMLocation": "Sophia",
  "InvoiceNumber": "Dell-264895-xp",
  "ErrorCode": 0
}
```

### /device/{deviceId}/windows/registry/key

This operation allows you to create a new Windows registry key.

#### Syntax

```
PUT /device/{deviceId}/windows/registry/key
```

#### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.



Name	Type	Mand.	Data Type	Description
DeviceRegistry Path	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	DeviceRegistryPath	The full path to the new registry key with the last element being the name of the new key. Use double slashes (//) for the path.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class *DeviceOperationResult*.

## Comments

Use double slashes (//) for the registry path.

Depending on the Windows operating system version accessing registry root keys like HKEY\_CLASSES\_ROOT, HKEY\_CURRENT\_USER, HKEY\_LOCAL\_MACHINE, HKEY\_USERS, HKEY\_CURRENT\_CONFIG HKLM might require further administrator privileges.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

### **/device/{deviceId}/windows/registry/value**

This operation allows you to create a new Windows registry key value.

### **Syntax**

PUT /device/{deviceId}/windows/registry/value

### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
DeviceRegistry PathValue	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http:// www.google.com or // myshare.myserver.com/ share1/ share01.</i> This parameter is only applicable for Quick Link rules.	DeviceRegistry PathValue	The full path to the Registry value with the last element being the new value's name.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class *DeviceOperationResult*.

## Comments

Use double slash (//) while providing registry path.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

### **/device/{deviceId}/inventory/patch/scan**

This operation launches a patch scan on a device.

#### **Syntax**

PUT /device/{deviceId}/inventory/patch/scan?forceupload=true&differential=true

#### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
forceupload	query	no	boolean	Define if the inventory is to be uploaded directly after generation.
differential	query	no	boolean	Defines if only those parts of the inventory are uploaded that have changed since the last upload.

## Returns

Returns the class [DeviceScanResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.



Error Code	Description
1	Operation failed.

### Example

The following example terminated skype, of which the id is 1968.

#### Request URL

```
https://localhost:1611/api/1/device/1000/inventory/patch/scan?forceupload=true&differencial=true
```

#### Response Body

```
{
  "ForceUpload": true,
  "DifferencialUpload": true,
  "ErrorCode": 0
}
```

### /device/{deviceId}/object/{objectId}/inst

This operation adds an instance for an inventory object type for a specific device, for example a new object type to the custom inventory.

### Syntax

```
PUT /device/{deviceId}/object/{objectId}/inst
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the device to which the object type instance is to be added.

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type of which an instance is to be added. This identifier is returned by one of the following operations: <i>GET /objects</i> or <i>PUT /object</i> .

## Returns

Returns the class *InventoryInstCreate*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.

Error Code	Description
16002	SQL execution error
16010	Login failed.
16040	Bad object type.

### Comments

Multiple values are not allowed.

### Example

This example adds a third instance of the object type 6113 to the custom inventory of the master device (ID 1000).

#### Request URL

```
https://localhost:1611/api/1/device/1000/object/6113/inst
```

#### Response Body

```
{
  "ObjectId": 1002,
  "ErrorCode": 0
}
```

whereby the `ObjectId` is the ID of the newly created instance. This ID is required when filling the attributes of the object type instance with device data.

### **/device/{deviceId}/object/{objectId}/inst/{instId}/attrs**

This operation allows you to specify the parameters values for an inventory object type instance of a device, for example the parameter values of a newly added custom inventory object type.

### Syntax

```
PUT /device/{deviceId}/object/{objectId}/inst/{instId}/attrs
```

The body must contain the definitions of the `PutObjectAttr` class.

```
{ "Name": "string", }
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the device to which the object type instance is to be added.

Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type. This identifier is returned by the <code>GET /objects</code> or <code>PUT /object</code> operation.

Name	Type	Mand.	Data Type	Description
instId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type instance for which the attribute values are to be specified. This identifier is returned by one of the following operations: GET /device/{deviceId}/object/{objectId}/inst or PUT /device/{deviceId}/object/{objectId}/inst.

Name	Type	Mand.	Data Type	Description
DeviceInventory Attrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	List[ <i>DeviceInventoryAttrs</i> ]	The list of attributes and their values that are to be specified for the respective object type instance and device.

## Returns

Returns the class *DeviceOperationResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.



Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.

## Comments

The body can contain as many attributes and values as there are attributes defined for the specified object type.

## Example

The following example defines the list of attributes that exist for object 6113 (custom inventory object type called "Location") and its third instance (ID: 1002).

### Request URL

```
https://localhost:1611/api/1/device/1000/object/6113/inst/1002/attrs
```

The body must contain the attribute values of the instance, for example:

```
{ "InstanceName": "Location
Weekdays", "Country": "France", "Town": "Nantes", "Department": "Accounting 157.EMEA", "Asset
Tag": "548925", "User ID": "678" }
```

### Response Body

```
{
  "ParentType": "_DB_OBJECTTYPE_DEVICE_",
  "ParentId": 1000,
  "ObjectType": 6113,
  "ObjectId": 1002,
  "ObjectAttrs": {
    "InstanceName": "Location Weekdays",
    "Country": "France",
    "Town": "Nantes",
    "Department": "Accounting 157.EMEA",
    "Asset Tag": "548925",
    "User ID": "678"
  },
  "ErrorCode": 0
}
```

## /device/{deviceId}/oprule/{opruleId}

This operation assigns an operational rule to a device.

## Syntax

```
PUT /device/{deviceId}/oprule/{opruleId}?fp=UsedInExtInt
```

Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
opruleid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the operational rule. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
fp	query	no	boolean	<p>Defines if the operational rule to create is to be used with integrated solutions. This parameter is only available if at least one external integration is defined.</p> <ul style="list-style-type: none"> <li>• <b>Available in integrated solutions:</b> The rule is to be available in integrated solutions and may be scheduled from the BCM console.</li> <li>• <b>Additional workflow needed:</b> The rule is to be available in integrated solutions and can only be used from there, that is, it is not schedulable from the BCM console.</li> <li>• <b>Not available in integrated solutions:</b> The rule is only used by the BCM agents.</li> </ul>

## Returns

Returns the class [DeviceOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.
16002	SQL execution error
16010	Login failed.
16051	The operational rule could not found.
16162	The operational rule is dependent on other rules that are not assigned to the device.
16222	The operational rule is not available for external integration.
16240	Device not found

## /device/{deviceId}/session

This operation open a new session on a device.

### Syntax

PUT /device/{deviceId}/session

The body must contain the definitions of the `credentials` class.

```
{ "credentials": { "user": "string", "password": "string" } }
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
credentials	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	<a href="#">SessionCredentials</a>	The user name and password of the BCM administrator with which to open the session on the device.

## Returns

Returns the class [SessionOperationResult](#).

## Comments

This ID is only valid as long as it continues to be used. After 5 minutes of inactivity it will expire

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

Error Code	Description
16002	SQL execution error
16010	Login failed.
16040	Bad object type.
16045	The device has no GUID.
16227	Remote credentials are required

### Example

This example opens a session on the master device as the `admin` user with the corresponding password `a`.

#### Request URL

```
https://localhost:1611/api/1/device/1000/session
```

The body must contain the login information, for example:

```
{ "credentials": { "user": "admin", "password": "a" } }
```

#### Response Body

```
{
  "session": {
    "id": "8ab7f262-b07d-8804-a3596c-ad092929b7"
  }
}
```

### /device/{deviceId}/fs/dir

This operation creates a new directory or renames an existing one under the specified path.

### Syntax

```
PUT /device/{deviceId}/fs/dir
```

The body must contain at least one of the definitions of the `DirectoryInfo` class.

```
{ "path": "string", "newpath": "string", }
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.



Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
rename	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	boolean	Select from this drop-down box the value false, if a new directory is to be created.
directoryInfo	query	no	DirectoryInfo	Enter the values for the new directory to create or the new name if it is to be renamed.

## Returns

Returns the class *DirectoryInfo*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
12	Access denied. Please verify your access rights.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

### Example

This example creates a new directory called "TestDir" under directory "D:\\Program Files":

#### Request URL

```
https://localhost:1611/api/1/device/1000/financial/summary?language=English
```

The body must contain the information for the new directory, for example:

```
{ "path": "D:\\\\Program Files\\TestDir" } }
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /device/{deviceId}/fs/file

This operation renames a file in specified path or copies it to a new location.

### Syntax

```
PUT /device/{deviceId}/fs/file
```

The body must contain the definitions of the `FileInfo` class.

```
{ "path": "string", "newpath": "string", }
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
rename	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	boolean	Set this parameter to true to rename the file, or to false to copy it.

Name	Type	Mand.	Data Type	Description
copy	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	boolean	Set this parameter to true to copy the file, or to false to rename it.
fileInfo	query	no	FileInfo	Enter the path values for the file to copy or rename to.

## Returns

Returns the class [FileInfo](#).

## Comments

You can perform either a copy or a rename operation at a time. If both flags are set to true an error will be generated.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
12	Access denied. Please verify your access rights.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.
14001	The file could not be found.

### Example

This example copies the file license.xml from its original location to the newly created directory "D:\\Program Files\\TestDir":

#### Request URL

```
https://localhost:1611/api/1/device/1000/fs/file?copy=true&rename=false
```

The body must contain the information for the new file location, for example:

```
{ "path": "D:\\\\Program Files\\license.xml", "newpath": "D:\\\\Program Files\\TestDir\\license.xml" }}
```

#### Response Body

```
{
  "errorCode": 0
}
```

## DELETE Operations

### /device/{deviceId}/process/{processIds}

This operation terminates one or more processes on a specific device.

### Syntax

```
DELETE /device/{deviceId}/process/{processIds}
```

### Parameters

The action takes the following parameters:



Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
processIds	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The list of process ids separated by comma(.). The process ids are returned by the GET /device/{deviceId}/processes operation.

## Returns

Returns the class *DeviceTerminateResult*.

## Comments

Multiple values of process ids are allowed, separated by comma.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

### Example

The following example terminated skype, of which the id is 1968.

#### Request URL

```
https://localhost:1611/api/1/device/1000/process/1968
```

#### Response Body

```
{
  "ProcessID": [
    "1968"
  ],
  "DACurrentConsoleUser": "admin",
  "ErrorCode": 0
}
```

### /device/{deviceId}

This operation deletes a specific device.

### Syntax

```
DELETE /device/{deviceId}
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

## Returns

Returns the class [DeviceOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.

Error Code	Description
16002	SQL execution error
16010	Login failed.
16040	Bad object type.
16042	The object can not be deleted as it is currently in use.
16046	The device is linked to objects.

### Example

The following example deletes device 1002.

#### Request URL

```
https://localhost:1611/api/1/device/1002
```

#### Response Body

```
{
  "ErrorCode": 0
}
```

### /device/{deviceId}/windows/registry/key

This operation allows you to delete a Windows registry key.

### Syntax

```
DELETE /device/{deviceId}/windows/registry/key
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
path	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The full path of the registry key to delete.



Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class [DeviceOperationResult](#).

## Comments

Use only single slashes (/) in the registry path.

Depending on the Windows operating system version accessing registry root keys like HKEY\_CLASSES\_ROOT, HKEY\_CURRENT\_USER, HKEY\_LOCAL\_MACHINE, HKEY\_USERS, HKEY\_CURRENT\_CONFIG HKLM might require further administrator privileges.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

### **/device/{deviceId}/windows/registry/value**

This operation allows you to delete a Windows registry value.

### **Syntax**

DELETE /device/{deviceId}/windows/registry/value

### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
path	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The full path of the registry key.

Name	Type	Mand.	Data Type	Description
<b>Name</b>	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The name of the registry value to delete.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

## Returns

Returns the class *DeviceOperationResult*.

## Comments

Use double slash (//) while providing registry path.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

### **/device/{deviceId}/object/{objectId}/inst/{instId}**

This operation deletes a specific instance of an inventory object type on a device, for example a third instance of a custom inventory object type on the device that has become obsolete.

### **Syntax**

DELETE /device/{deviceId}/object/{objectId}/inst/{instId}

### **Parameters**

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the device on which the object type instance is to be deleted.



Name	Type	Mand.	Data Type	Description
objectId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type whose instance is to be deleted. This identifier is returned by one of the following operations: <code>GET /objects</code> or <code>PUT /object</code> .

Name	Type	Mand.	Data Type	Description
instId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the object type instance to be deleted. This identifier is returned by one of the following operations: GET /device/{deviceId}/object/{objectId}/inst or PUT /device/{deviceId}/object/{objectId}/inst.

## Returns

Returns the class *DeviceOperationResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16010	Login failed.
16040	Bad object type.
16047	This object is used by a query and cannot be deleted.

### Example

This example deletes an custom inventory object type called *Location* (ID: 6113).

#### Request URL

```
https://localhost:1611/api/1/device/1000/object/6113/inst/1002
```

#### Response Body

```
{
  "ParentType": "_DB_OBJECTTYPE_DEVICE_",
  "ParentId": 1000,
  "ObjectTypeId": 6113,
  "ObjectId": 1002,
  "ErrorCode": 0
}
```

whereby the `ParentId` is the ID of the device (master), the `ObjectTypeId` is the ID of the object type (Location) the `ObjectId` is the ID of the deleted object type instance.

### /device/{deviceId}/fs/dir

This operation deletes a specific directory.

### Syntax

```
DELETE /device/{deviceId}/fs/dir?path=DirPath
```

### Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.

Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
path	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	boolean	The complete path of the directory to be deleted. Use double backslashes (\\) in the path.

## Returns

Returns the class [DeviceDeleteDir](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.

## Example

This example deletes the directory "TestDir" under "D:\\Program Files":

### Request URL

```
https://localhost:1611/api/1/device/1000/fs/dir?path=D%3A%5C%5C%5CProgram+Files%5C%5CTestDir
```

### Response Body

```
{
  "errorCode": 0
}
```

## /device/{deviceId}/fs/file

This operation renames a file in specified path or copies it to a new location.

## Syntax

```
DELETE /device/{deviceId}/fs/file?path=FilePath
```

## Parameters

The action takes the following parameters:

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.



Name	Type	Mand.	Data Type	Description
Session-ID	header	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The unique identifier of the session. This identifier is recovered by executing the <code>PUT /device/{deviceId}/session</code> operation.

Name	Type	Mand.	Data Type	Description
path	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	boolean	The complete path of the file to be deleted. Use double backslashes (\\) in the path.

## Returns

Returns the class [DeviceDeleteFile](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
12	Access denied. Please verify your access rights.

Error Code	Description
8001	Acknowledgment is required.
8002	The acknowledgment was refused by the end user.
8003	The acknowledgment has expired.
14001	The file could not be found.

## Example

This example deletes the file "license.xml" under the "D:\\Program Files\\TestDir" directory:

### Request URL

```
https://localhost:1611/api/1/device/1000/fs/file?path=D%3A%5C%5C%5CProgram+Files%5C%5CTestDir%5C%5Clicense.xml
```

### Response Body

```
{
  "errorCode": 0
}
```

## Device Return Classes

The following classes can be returned by the different device actions:

Class	Description
<i>CatalogKeywordTranslation</i>	Returns the translated keywords for the Software Catalog.
<i>DeviceAssetCatalog</i>	Return the Software Catalog List.
<i>DeviceAssetCatalogSummary</i>	Return the Software Catalog List and operation result.
<i>DeviceAssetFinancial</i>	Return the available financial data.
<i>DeviceAssetFinancialSummary</i>	Return the financial summary and operation result.
<i>DeviceAssetHardware</i>	Return the available hardware data.
<i>DeviceAssetHardwareSummary</i>	Return the hardware summary and operation result.
<i>DeviceAssetSoftware</i>	Return the available software data.
<i>DeviceAssetSoftwareSummary</i>	Return the software summary and operation result.
<i>DeviceAssetSummary</i>	Return the available general device data.

Class	Description
<i>DeviceAssetSummaryResult</i>	Return the general device summary and operation result.
<i>DeviceAttributes</i>	Return information about all attributes of a device.
<i>DeviceAttributesInfo</i>	Return all attributes of a device.
<i>DeviceCategoryResultValues</i>	Return the Windows event categories of a device.
<i>DeviceCategoryValues</i>	Return the information about a specific event category.
<i>DeviceCreate</i>	Provides necessary information for creating a new device.
<i>DeviceDeleteDir</i>	The result of the directory delete operation.
<i>DeviceDeleteFile</i>	The result of the file delete operation.
<i>DeviceData</i>	Set the name of the new device.
<i>DeviceExit</i>	Return device shutdown result.
<i>DeviceExitInfo</i>	Returns device shutdown result.
<i>DeviceFileSystem</i>	Returns the list of drives of a device or the files and folders of a folder.
<i>DeviceFileSystemDir</i>	Provides information about a specific child.
<i>DeviceFileSystemDirInfo</i>	Returns the list of children of a specific directory.
<i>DeviceFileSystemInfo</i>	Returns the list of drives of a device.
<i>DeviceFinancialAttributes</i>	Return all financial attributes and their values of a specific device.
<i>DeviceInventoryAttrValue</i>	Shows the list of attributes and their values of the object type.
<i>DeviceInventoryAttrs</i>	Provide the attribute parameters and their values to be specified.
<i>DeviceInventoryInsts</i>	Return the instances of a specific object type on a given device.
<i>DeviceOperationResult</i>	Return the final execution status.
<i>DeviceProcesses</i>	Get processes of a device.
<i>DeviceProcessValues</i>	Provide information about a specific process.
<i>DeviceReadFile</i>	Returns information about a specific file and its content.
<i>DeviceRegistryKey</i>	Provide information about a specific key.
<i>DeviceRegistryKeys</i>	Return the list of Registry keys.

Class	Description
<i>DeviceRegistryPath</i>	Provide the new Registry key path.
<i>DeviceRegistryPathValue</i>	Provide the data for the new key value.
<i>DeviceRegistryValue</i>	Provide information about a specific value.
<i>DeviceRegistryValues</i>	Return the list of Registry values.
<i>DeviceResult</i>	Return ping operation result.
<i>DeviceScan</i>	Provide new Patch inventory upload criteria.
<i>DeviceScanResult</i>	Return patch inventory scan result.
<i>DeviceService</i>	Provide information about a specific Windows service.
<i>DeviceServiceInfo</i>	Return the list of Windows services.
<i>DeviceTerminateResult</i>	Return process termination result.
<i>DeviceUpdate</i>	Provide all parameters that require updating on the device.
<i>DeviceWindowsEvent</i>	Return information about a specific event.
<i>DeviceWindowsEventResult</i>	Return the list of events.
<i>DirectoryInfo</i>	Contains directory information.
<i>FileInfo</i>	Contains information for a specific file.
<i>FilteredActionsResult</i>	Return the list of actions with their availability status.
<i>FilteredActionsValues</i>	Return the translated keywords.
<i>HardwareKeywordTranslation</i>	Return the hardware inventory keywords in the target language.
<i>InventoryInstCreate</i>	Return the object type addition result.
<i>JnlpResult</i>	The XML content of the JNLP file.
<i>KeywordTranslation</i>	An array returning the keywords translated to the target language.
<i>OpRuleHistoryResult</i>	Return list of assigned operational rules,
<i>SessionOperationResult</i>	Returns the open session operation result.
<i>SessionResult</i>	Return the ID of a session.
<i>SoftwareKeywordTranslation</i>	Return software inventory keywords translated to target language.

### CatalogKeywordTranslation

This class returns the keywords for the returned software catalog inventory translated to the required language.

Parameter	Type	Description
<b>Name</b>	string	The name of the application as extracted from the installed software translation file. This is usually the complete product name, not including manufacturer or version information, for example, FinePrint 2000, InstallShield, Adobe Acrobat.
Version	string	The version string for the application as extracted from the translation file. This is a text string and contains generally a mixture of digits and letters as required by the entry, for example, 3.0 Rev. B.
Software Manufacturer	string	The name of the manufacturer of the application as extracted from the translation file. This is a free-form field and may contain the name by which the company is known or its complete registered name, such as Microsoft Corporation, Adobe Systems Incorporated, BMC Software, Inc.
Type	string	This field defines the type of the software, which may be Application, Web Browser, Communications, and so on.
Category	string	The category that the software belongs to, such as <i>Desktop Publishing</i> , <i>Development Environment</i> , <i>Network Monitoring</i> , and so on.

### DeviceAssetCatalog

This class returns the list of all software that is installed on a specific device and is included in the Software Catalog.

Parameter	Type	Description
<b>Name</b>	string	The name of the application as extracted from the installed software translation file. This is usually the complete product name, not including manufacturer or version information, for example, FinePrint 2000, InstallShield, Adobe Acrobat.
Version	string	The version string for the application as extracted from the translation file. This is a text string and contains generally a mixture of digits and letters as required by the entry, for example, 3.0 Rev. B.

Parameter	Type	Description
Software Manufacturer	string	The name of the manufacturer of the application as extracted from the translation file. This is a free-form field and may contain the name by which the company is known or its complete registered name, such as Microsoft Corporation, Adobe Systems Incorporated, BMC Software, Inc.
Type	string	This field defines the type of the software, which may be Application, Web Browser, Communications, and so on.
Category	string	The category the software belongs to.
KeywordTranslation	object[ <a href="#">CatalogKeywordTranslation</a> ]	An array returning the keywords translated to the target language.

### DeviceAssetCatalogSummary

This class returns the result of a software catalog summary operation, that is the list of installed software and the operation result.

Parameter	Type	Description
_DB_OBJECTTYPCLASS_SOFTWAREINVENTORY_	LIST[ <a href="#">DeviceAssetCatalog</a> ]	Provides the list of all software catalog entries, one list entry per software entry.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceAssetFinancial

This class returns all the available financial data of a specific device.

Parameter	Type	Description
PurchaseCostofAsset	float	Enter into this field the total purchasing costs of the asset.
ServiceStartDate	string	Select in this field the date at which the asset went online, that is, was finally up and running. This date must be later than the <b>Date Received</b> value. You can clear the field again by clicking the <b>Erase</b> icon to the right of the field.
SupportProvider	string	Enter the name company that provides the support for this asset.
SupportExpirationDate	string	The date at which the support contract for the asset runs out.

Parameter	Type	Description
WarrantyExpirationDate	string	Select in this field the date at which the warranty for the asset expires. This date must be later than the <b>Service Start Date</b> value. You can clear the field again by clicking the <b>Erase</b> icon to the right of the field.
ResidualValue	float	Enter into this field the currently remaining value of the asset. The residual value is an estimate of the value of the asset at the time it is sold or disposed of; it may be zero. Residual value is also known as scrap value or salvage value.
ActualCost	float	The cost after the depreciation calculation.
KeywordTranslation	object[ <a href="#">KeywordTranslation</a> ]	An array returning the keywords translated to the target language.

### DeviceAssetFinancialSummary

This class returns a summary of all the financial data that is available for a specific device.

Parameter	Type	Description
Summary	object[ <a href="#">DeviceAssetFinancial</a> ]	Provides the list of all available financial data of a specific device in the form of a <code>DeviceAssetFinancial</code> object.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceAssetHardware

This class returns all the available hardware data of a specific device.

Parameter	Type	Description
DeviceIPAddress	string	The IP address of the autodiscovered device.
MacAddress	string	The MAC (hardware) address of the device.
SerialNumber	string	The serial number of the device.
MostUsedUser	string	The User who uses the device most often.
Type	string	Defines the type of event to be monitored, the values depend on the source chosen above.



Parameter	Type	Description
DeviceName	string	The name of the device to which the alert is related.
LastUpdate	string	The date and time at which the patch inventory was last updated on the device to check if all its patches are up to date.
OperatingSystemName	string	The operating system name of the scanned device.
TopologyType	string	The topology type of the device, that is, if the managed device is a master, a relay or a simple client. It may also be an unconnected, a scanned, a deprecated or an unknown device.
DomainName	string	The full name of the device including the domain it belongs to, i.e., <code>kirk.enterprise.starfleet.com</code> .
Domain	string	Enter into this field the name of the domain the new device should belong to, for example, <i>TESTLAB</i> . If you entered a name for the workgroup above the domain value prevails.
Model	string	The computer system model.
_DB_ATTR_VIMVENDOR_	string	Defines the type of virtual machine running on the host, that is, the name of the software used. This may be either <i>None</i> if no virtual machine is installed on the device, <i>VM Ware Server</i> , <i>Microsoft VirtualPC Server</i> , <i>VirtualBox</i> or <i>Parallels</i> .
_DB_ATTR_HYPERVISOR_VENDOR_	string	Displays if the agent device hosts a hypervisor, in which case this field displays the name of the virtualizing software, otherwise it is empty.
AssetUser	string	Click the <b>Add User</b> icon to the right of the field to select the owner of this asset.
AssetAdmin	string	Click the <b>Add Administrator</b> icon to the right of the field to select the support contact who is responsible for this asset.
KeywordTranslation	object[ <a href="#">HardwareKeywordTranslation</a> ]	An array returning the keywords translated to the target language.

### DeviceAssetHardwareSummary

This class returns the result of a financial summary operation, that is available hardware data and the operation result.

Parameter	Type	Description
Device	LIST[ <a href="#">DeviceAssetHardware</a> ]	Returns the search result for one specific device.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceAssetSoftware

This class returns all the available software data of a specific device.

Parameter	Type	Description
<b>Name</b>	string	The name of the application as extracted from the installed software translation file. This is usually the complete product name, not including manufacturer or version information, for example, FinePrint 2000, InstallShield, Adobe Acrobat.
Version	string	The version string for the application as extracted from the translation file. This is a text string and contains generally a mixture of digits and letters as required by the entry, for example, 3.0 Rev. B.
Software Manufacturer	string	The name of the manufacturer of the application as extracted from the translation file. This is a free-form field and may contain the name by which the company is known or its complete registered name, such as Microsoft Corporation, Adobe Systems Incorporated, BMC Software, Inc.
Type	string	This field defines the type of the software, which may be Application, Web Browser, Communications, and so on.
Installation Directory	string	The fields of this column display the full path of the respective installation directories.
KeywordTranslation	object[ <a href="#">SoftwareKeywordTranslation</a> ]	An array returning the keywords translated to the target language.

### DeviceAssetSoftwareSummary

This class returns the result of a financial summary operation, that is available software data and the operation result.

Parameter	Type	Description
_DB_OBJECTTYPCCLASS _SOFTWAREINVENTORY_	LIST[ <a href="#">DeviceAssetSoftware</a> ]	Provides the list of all software catalog entries, one list entry per software entry.

Parameter	Type	Description
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceAssetSummary

This class returns all the available general data of a specific device.

Parameter	Type	Description
deviceId	string	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
DEVICENAME	string	The name of the device under which it is known in the BCM database.
IPADDRESS	string	The IP address of the device in its dotted version, such as <i>194.50.68.255</i> or <i>2001:db8:85a3::8a2e:370:7334</i> .
MACADDRESS	string	The MAC (hardware) address of the device.
TOPOLOGYTYPE	string	The topology type of the device, that is, if the managed device is a master, a relay or a simple client. It may also be an unconnected, a scanned, a deprecated or an unknown device.
VIMMACHINEVENDOR	string	Defines the type of virtual machine running on the host, that is, the name of the software used.
HYPERVISORVENDOR	string	Defines if the agent device hosts a hypervisor, in which case this parameter must contain the name of the virtualizing software, otherwise it must be left empty.
MANUFACTURER	string	The name of the manufacturer of the device.
MODEL	string	The computer system model.
SERIALNUMBER	string	The serial number of the hard disk of the device.
LASTUPDATE	string	The date and time at which the device information was last updated.
TOTALDISK	string	The total size of all disks of the device.
LOGICALDISKSSPACE	string	The list of all sizes of all logical disks, separated by a colon (,).
LOGICALDISKSFREESPACE	string	The list of all free space of all logical disks, separated by a colon (,).
PROCESSOR	string	The description of the processor type of the device.

Parameter	Type	Description
VIDEOCONTROLLER	string	The description of the video controller.
TOTALMEMORY	string	The total memory of the device.
NBOFPROCESSORS	string	The total number of processors of the device.
PHYSICALMEMORY	string	The total physical memory of the device.
NETWORKADAPTER	string	The list of all network adapters, of which either the IP address is not empty and the <b>Network Adapter Configuration Description</b> is not equal to <i>Virtual</i> or <i>Loopback</i> .
DOMAINNAME	string	The full name of the domain the currently selected device belongs to, that is, <i>kirk.enterprise.starfleet.com</i> .
OPERATINGSYSTEMNAME	string	The name of the operating system installed on the currently selected device.
OPERATINGSYSTEMVERSION	string	The version of the operating system composed of its major, minor and build number.
LASTLOGGEDINUSER	string	The user that is currently logged on to the device or if no user is logged on the user that was last logged on to the device.

### DeviceAssetSummaryResult

This class returns the result of a device summary operation, that is available general data and the operation result.

Parameter	Type	Description
Summary	object[ <a href="#">DeviceAssetSummary</a> ]	Provides the list of all available general data of a specific device in the form of a <code>DeviceAssetFinancial</code> object.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceAttributes

This class provides all attributes and their values that are to be updated on a specific device.

Parameter	Type	Description
_DB_ATTR_OSD_PARENT_	integer	The name or the IP address of the parent of the device. In case of the master or unconnected devices this field is empty.
_DB_ATTR_SMALLICON_	string	The name and relative path of the small icon file to be used to represent this device.

Parameter	Type	Description
_DB_ATTR_LARGEICON_	string	The name and relative path of the large icon file to be used to represent this device.
_DB_ATTR_ENTRYDN_	string	The base DN is the start entry in the directory tree. You can enter this value either in the LDAP notation or as UNC. For example for an Active Directory domain with the name <i>kirk.bridge.enterprise.com</i> , this entry would look like this: <i>dc=kirk, dc=bridge, dc=enterprise, dc=com</i> in <b>LDAP notation</b> or <i>kirk.bridge.enterprise.dc=com</i> as <b>UNC</b> .
_DB_ATTR_DEV_GUID_	string	The unique identifier of the device assigned to it by BCM.
_DB_ATTR_DEV_RETIREDBY_	string	The name of the administrator who deprecated the device.
_DB_ATTR_DEVICE_FAMASSETTAG_	string	Enter the unique identifier of the asset. The identifier can have no more than 64 characters.
_DB_ATTR_DEVICE_PRIMARYUSER_	string	The name of the primary user in the format of DOMAIN/USERNAME.
_DB_ATTR_NAME_	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
UserName	string	The login name of the user who is either currently connected to the device, or, if no user is connected, the user who last connected to the device.
_DB_ATTR_DEV_TYPE_	string	The type of the device, that is, which purposes the device server, if it is a server, a workstation, a printer or a game console, etc. You can manually modify this value. However, in this case you also need to deactivate the automatic updates, otherwise the device type reverts to its original type at the next update. To switch to manual update click the icon next to the box, which appears when you manually change the type.
FAMLocation	string	The country/region/town/building/geographical area at which the asset is located.
_DB_ATTR_VIMVENDOR_	string	Defines the type of virtual machine running on the host, that is, the name of the software used. This may be either <i>None</i> if no virtual machine is installed on the device, <i>VM Ware Server</i> , <i>Microsoft VirtualPC Server</i> , <i>VirtualBox</i> or <i>Parallels</i> .
_DB_ATTR_HYPERVISORVENDOR_	string	Displays if the agent device hosts a hypervisor, in which case this field displays the name of the virtualizing software, otherwise it is empty.
_DB_ATTR_HYPERVISORVERSION_	string	The version number of the hypervisor.

Parameter	Type	Description
_DB_ATTR_DEV_DOMAINNAME_	string	The full name of the domain the currently selected device belongs to, that is, <i>kirk.enterprise.starfleet.com</i> .
_DB_ATTR_DEV_NETWORKNAME_	string	The network name of the machine, either as its short or complete network name, for example, <i>scotty</i> or <i>scotty.enterprise.com</i> , or as its IP address in dotted notation, for example, <i>194.45.245.5</i> or <i>2001:db8:85a3::8a2e:370:7334</i> .
_DB_ATTR_DEV_HTTPPORT_	integer	The range of ports on which the HTTP server listens for and sends data from.
_DB_ATTR_DEV_HTTPCONSOLEPORT_	integer	The number of the port that the console uses for communication with the agent.
_DB_ATTR_DEV_SSL_	string	Defines if the agent sends any communication in secure format.
_DB_ATTR_DEVICE_NETBIOSNAME_	string	The NetBIOS name of the client. For managed devices which have Linux or MAC OS as their operating system this entry will not be displayed.
_DB_ATTR_DEVICE_IPADDRESS_	string	The IP address of the device in its dotted version, such as <i>194.50.68.255</i> or <i>2001:db8:85a3::8a2e:370:7334</i> .
_DB_ATTR_DEVICE_NETWORKMASK_	string	The subnet mask of the device in its dotted version, such as <i>255 . 255 . 255 . 0</i> .
_DB_ATTR_DEVICE_UNDERNAT_	boolean	Indicates if at least one piece of hardware of the device uses network address translation.
_DB_ATTR_DEVICE_MACADDRESS_	string	The MAC (hardware) address of the device.
_DB_ATTR_DEVICE_HOSTID_	string	The host ID if the operating system of the currently selected client is Linux or MAC OS, if the operating system is Window it must either be the asset tag or the BIOS serial number depending on the manufacturer of the client in the respective format.
_DB_ATTR_DEVICE_DISKSERNUM_	string	The serial number of the hard disk of the device.
_DB_ATTR_PARENT_	string	The name or the IP address of the parent of the device. In case of the master or unconnected devices this field is empty.
_DB_ATTR_DEVICE_TOPOLOGYTYPE_	string	The topology type of the device, that is, if the managed device is a master, a relay or a simple client. It may also be an unconnected, a scanned, a deprecated or an unknown device.
_DB_ATTR_DEV_OSNAME_	string	The name of the operating system installed on the currently selected device.

Parameter	Type	Description
_DB_ATTR_DEV_OSVERMAJOR_	integer	The major version number of the operating system installed on the device.
_DB_ATTR_DEV_OSVERMINOR_	integer	The minor version number of the operating system installed on the device.
_DB_ATTR_DEV_OSREVISION_	string	The revision number of the operating system installed on the currently selected device.
_DB_ATTR_DEV_OSBUILD_	integer	The build number of the operating system installed on the device.
_DB_ATTR_DEV_AGENTVERMAJOR_	integer	The major version number of the agent installed on the device.
_DB_ATTR_DEV_AGENTVERMINOR_	integer	The minor version number of the agent installed on the device.
_DB_ATTR_DEV_AGENTREVISION_	string	The revision number of the agent installed on the device.
_DB_ATTR_DEV_AGENTBUILD_	string	The build number of the agent installed on the device.
_DB_ATTR_PATCHCONFIGFILEVERSION_	string	The currently installed version of the configuration files of the Patch Management functionality.
_DB_ATTR_PKGFACTORY_	boolean	Indicates if the currently selected device is a Packager in the Package Factory, that is, if packages can be created on it.
_DB_ATTR_PATCHPROXY_	boolean	Indicates if the currently selected device is serving as a Patch Manager, that is, if it can handle MS Secure files and all other options pertaining to patch management.
_DB_ATTR_OSD_	boolean	Indicates if the currently selected device is a OSD Manager, that is, if it can create and manage operating system deployments as well as install them on the defined target devices.
_DB_ATTR_VPROAVAILABLE_	boolean	Indicates if the device is equipped with Intel's vPro firmware.
_DB_ATTR_RISCANNER_	boolean	Defines if the device acts as an Asset Discovery Scanner in your network.
_DB_ATTR_ROLLOUTSRV_	boolean	Defines if the device acts as a rollout server for BCM agent deployment.
_DB_ATTR_DEV_UPDATETIME_	string	The date and time at which the device information was last updated.
_DB_ATTR_CREATEDBY_	string	The name of the creator of the object such as an administrator.

Parameter	Type	Description
_DB_ATTR_CREATETIME_	string	The date and time at which the object was originally created.
_DB_ATTR_LASTMODIFIEDBY_	string	Displays the name of either the last person that last modified the object or its contents, such as the administrator, or it may be the system that last executed any modifications.
_DB_ATTR_LASTMODIFYTIME_	string	The date and time of the last modification of the object.
_DB_ATTR_NOTES_	string	Free text field that may be edited to display general information about the object and its contents.

### DeviceAttributesInfo

This class returns the attributes found for a specific device.

Parameter	Type	Description
Values	LIST[ <a href="#">DeviceAttributes</a> ]	This list returns one entry for each of the found attributes.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceCategoryResultValues

This class returns the list of and information about the Windows event categories that are found on a device.

Parameter	Type	Description
DeviceCategoryResultValues	LIST[ <a href="#">DeviceCategoryValues</a> ]	Return the Windows event categories of a device.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceCategoryValues

This class returns the available information about a specific Windows event category found on the device.

Parameter	Type	Description
NtEventCategoryName	string	The name of the event category.
NtEventCategoryType	string	The type of the event category
NtEventCategoryDesc	string	A description of the event category.
NtEventCategorySize	integer	The size of the category.



### DeviceCreate

This class returns the operation result and if successful the ID of the newly created device.

Parameter	Type	Description
objectId	string	The unique BCM database ID of the object.
errorCode	string	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceData

This class defines the name of the new device to create.

Parameter	Type	Description
<b>Name</b>	string	The user-friendly name under which the new device is to be created in the AssetCore database.

### DeviceDeleteDir

This class contains result of the directory delete operation.

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### DeviceDeleteFile

This class contains result of the file delete operation.

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### DeviceExit

This class returns the result of a device shutdown or reboot operation.

Parameter	Type	Description
deviceId	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.

Parameter	Type	Description
HowToExit	string	Shows the type of shutdown operation executed on the device, that is, if it was completely shut down or rebooted.

### DeviceExitInfo

This class returns the result of a device shutdown or reboot operation.

Parameter	Type	Description
DeviceExitInfo	Array[ <a href="#">DeviceExit</a> ]	Return device shutdown result.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceFileSystem

This class returns the available information about a specific device drive.

Parameter	Type	Description
name	string	The name of the drive.
driveType	string	The type of drive.
volumeName	string	The volume name of drive.

### DeviceFileSystemDir

This class provides information about a specific child of a directory.

Parameter	Type	Description
name	string	The name of the directory or file.
isDir	boolean	Indicates if it is a directory or a file.
"modifiedDate	string	The date and time in the form of a timestamp at which the file/directory was last modified.
"mode	integer	Mode information of directory/file.
"isText	boolean	Indicates if the file is a text or any other type of file. This parameter is not applicable to directories.
"isHidden	boolean	Indicates if the file is a hidden file. This parameter is not applicable to directories.

### DeviceFileSystemDirInfo

This class returns the list of children of a specific directory.

Parameter	Type	Description
path	LIST[ <a href="#">DeviceFileSystemDir</a> ]	Provides information about a specific child.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### DeviceFileSystemInfo

This class returns the list of drives of a device.

Parameter	Type	Description
paths	LIST[ <a href="#">DeviceFileSystem</a> ]	Returns information about a specific drive.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### DeviceFinancialAttributes

This class provides the financial attributes and their values that are to be set on a specific device.

This return class has the following parameters:

Parameter	Type	Description
FAMID	integer	The unique BCM database identifier of the financial entry of a device.
LifeCycleStatus	string	<p>Select the lifecycle status from the list box. The predefined values are:</p> <ul style="list-style-type: none"> <li>• <b>On Order:</b> The asset was already ordered but has not yet arrived at its destination.</li> <li>• <b>Received:</b> The asset arrived at its destination.</li> <li>• <b>In Stock:</b> The asset has arrived at its destination and is currently located in the warehouse.</li> <li>• <b>Deployed:</b> The asset is installed at its final destination.</li> <li>• <b>Retired:</b> The asset has arrived at the end of its lifecycle and was removed from the network.</li> </ul> <p>If you have defined your own custom status values these are also available in this list.</p>
AssetUser	string	The owner of this asset.

Parameter	Type	Description
AssetAdmin	string	The support contact who is responsible for this asset.
Department	string	The department that is responsible for this device. The department is represented by a user group.
Vendor	string	Enter the name of the vendor of the asset, for example, <i>Hewlett Packard</i> or <i>Dell</i> .
VendorSKU	string	The unique identifier of this asset as assigned by its vendor (stock-keeping unit).
PONumber	string	Enter the number of the purchase order for the asset.
WarrantyExpirationDate	string	Select in this field the date at which the warranty for the asset expires. This date must be later than the <b>Service Start Date</b> value. You can clear the field again by clicking the <b>Erase</b> icon to the right of the field.
WarrantyCost	integer	Enter the total cost for the warranty contract of the asset.
SupportCost	integer	Enter the total cost for the support contract of the asset.
SupportProvider	string	Enter the name company that provides the support for this asset.
SupportProviderPhoneNumber	string	Enter the phone number under which your direct contact at the support provider can be reached.
PurchaseDate	string	Select in this field the purchase date of the asset. You can clear the field again by clicking the <b>Erase</b> icon to the right of the field.
DateReceived	string	Select in this field the date at which the asset arrived at its destination. This date must be later than the <b>Purchase Date</b> value. You can clear the field again by clicking the <b>Erase</b> icon to the right of the field.

Parameter	Type	Description
ServiceStartDate	string	Select in this field the date at which the asset went online, that is, was finally up and running. This date must be later than the <b>Date Received</b> value. You can clear the field again by clicking the <b>Erase</b> icon to the right of the field.
PurchaseCostofAsset	integer	Enter into this field the total purchasing costs of the asset.
ResidualValue	integer	Enter into this field the currently remaining value of the asset. The residual value is an estimate of the value of the asset at the time it is sold or disposed of; it may be zero. Residual value is also known as scrap value or salvage value.
UsefulLife	integer	Enter into this field the total amount of time that the asset is expected to be up and working in your network in months from the service start date onwards.
LeaseCostofAsset	integer	Enter into this field the total costs for the lease of the asset.
LeaseTerm	integer	Select from this list the time in months for the leasing contract.
IsPurchased	boolean	Indicates if the device was purchased or leased, <code>true</code> if purchased, <code>false</code> if leased.
DepreciationType	string	The method used to calculate the costs: <ul style="list-style-type: none"> <li>• <b>Straight Line:</b> the most often used method, in which the company estimates the residual value of the asset at the end of the period during which it is used to generate revenues (useful life) and expenses a portion of <i>original cost</i> in equal increments over that period.</li> <li>• <b>Declining Balance:</b> the book value is multiplied by a fixed rate.</li> </ul>
SupportExpirationDate	string	The date at which the support contract for the asset runs out.
FAMLocation	string	The country/region/town/building/geographical area at which the asset is located.

Parameter	Type	Description
AssetTag	string	Enter the unique identifier of the asset. The identifier can have no more than 64 characters.
InvoiceNumber	string	Enter into this field the number of the invoice.
InvoiceDate	string	Enter into this field the date of the invoice. The date must be equal to or later than the <b>Purchase Date</b> .
RestrictLCSUpdate	boolean	This parameter defines if the lifecycle status value is updated automatically or manually. Activate this parameter to manually define the lifecycle status of a device. In this case the automatic evaluation of the financial information of the device is suspended.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceInventoryAttrValue

This class shows the list of attributes and their values of the object type.

Parameter	Type	Description
AttrName	string	The attribute name and its parameter value. Depending on the attribute there might be more than one entry.

### DeviceInventoryAttrs

This class provides the attribute parameters and their values that be specified for the object type on the respective device, for example the location device data for a "Location" custom inventory object type.

Parameter	Type	Description
<b>Name</b>	string	The name of the attribute and the value it is to have. Be careful to exactly write the attribute name as it is defined, paying attention to small and capital letters, otherwise the data will not be set. The list of attributes of a specific object type is returned by the <code>GetObjectAttrsResult</code> class of the <code>GET /object/{objectId}/attrs</code> operation.

## DeviceInventoryInsts

This class returns the instances of a specific object type on a given device. It displays the following information:

Parameter	Type	Description
ObjectId	integer	The unique BCM database identifier of the respective device.
ObjectName	string	This parameter contains the list of object instances of the object type. It will display one <code>DeviceInventoryAttrValue</code> class for each instance.
Object	List[ <a href="#">DeviceInventoryAttrValue</a> ]	This object contains all the parameters that require updating on the device.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

## DeviceOperationResult

This class returns the final status of the executed operation, that is if it was successful or not, and if not the corresponding error message.

Parameter	Type	Description
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

## DeviceProcesses

This operation returns the list of processes running on a specific device.

Parameter	Type	Description
DeviceProcesses	List[ <a href="#">DeviceProcessValues</a> ]	A list of <code>DeviceProcessValues</code> entries, providing individual information about a specific process each.
Total	integer	The total number of devices found matching the search query.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

## DeviceProcessValues

This class provides information about a specific process.

Parameter	Type	Description
ProcessOwner	string	The name of the owner of the process, which in most cases is either SYSTEM for the processes started by the system or the login name of the user.
ProcessName	string	The name of the process.
ProcessPath	string	The full path to the executable file of the process.
ProcessTotalCpu	string	The total processor time in seconds used by a process since it was started.
ProcessCurrentCpu	integer	The percentage of time that a process used the CPU since the last update.
ProcessID	string	The numerical identifier that uniquely distinguishes a process while it runs.
ProcessParentID	string	The PID of the parent process of the currently selected process is a child. The field is empty if it is not a child process.
ProcessMemoryVirtual	integer	The amount of virtual memory or address space committed to a process. Not applicable for Linux and Mac OS.
ProcessMemoryUsed	integer	The current working set of a process in KB. The current working set is the number of pages currently resident in the memory.
ProcessThreadsCount	string	The number of threads running in a process. Not applicable for Linux and Mac OS.

### DeviceReadFile

This class returns information about a specific file and the requested content.

Parameter	Type	Description
paths	string	The path of the file to be read.
offset	integer	The offset from where on to start displaying the content.
length	integer	The number of characters read from the file from the defined offset onwards.
result	string	The content of the file from the defined offset onwards.



### DeviceRegistryKey

This class returns the information about a specific registry key.

Parameter	Type	Description
<b>Name</b>	string	The name of the registry key.

### DeviceRegistryKeys

This class returns the list of all registry keys of a specific key on a device.

Parameter	Type	Description
DeviceRegistryKeys	Array[ <a href="#">DeviceRegistryKey</a> ]	Provide information about a specific key.

### DeviceRegistryPath

This class provides the full path and name for the registry key to be created.

Parameter	Type	Description
path	string	The complete path to the registry key.

### DeviceRegistryPathValue

This class provides the data for the new Registry key value.

Parameter	Type	Description
path	string	The full path to the new value.
<b>Name</b>	string	The name of the registry value.
type	string	The type of the registry value.
value	string	The actual data of the registry value.

### DeviceRegistryValue

This class returns the information about a specific registry value.

Parameter	Type	Description
<b>Name</b>	string	The name of the registry value.
Type	string	The type of the registry value.
Data	string	The actual data of the registry value.

### DeviceRegistryValues

This class returns the list of all registry values of a specific key.

Parameter	Type	Description
DeviceRegistryValues	Array[ <a href="#">DeviceRegistryValue</a> ]	Return the list of Registry values.

### DeviceResult

This class returns the result of a ping operation.

Parameter	Type	Description
Result	integer	Return the number of pings sent before the device answered.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceScan

This class shows the upload criteria for the newly generated patch inventory.

Parameter	Type	Description
ForceUpload	boolean	Specifies if the results of the forced scan are to be uploaded directly or at the regular scheduled time.
DifferentialUpload	boolean	Specifies if the inventory is to be completely replaced which each upload when differences are detected or only the delta, that is, the modifications of the inventory. If the inventory template is changed the next inventory will always be a complete inventory, even if this option is activated.

### DeviceScanResult

This class returns the result of a patch inventory scan on a specific device.

Parameter	Type	Description
DeviceScanResult	object[ <a href="#">DeviceScan</a> ]	Return patch inventory scan result.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

## DeviceService

This class provides information about a specific Windows service.

Parameter	Type	Description
displayName	string	A explicit name of the service name, that is, <i>Windows Update</i> , instead of <i>wuauserv</i> .
serviceName	string	The internal name of the service.
state	string	The current status of the service.
Description	string	A description of the service.

## DeviceServiceInfo

This class provides the list of all Windows services on a specific device.

Parameter	Type	Description
DeviceServiceInfo	Array[ <a href="#">DeviceService</a> ]	Provide an array of <a href="#">DeviceService</a> objects, each of which provides information about a specific service.
Total	integer	The total number of services found on the device.

## DeviceTerminateResult

This class returns if one or more processes could be terminated.

Parameter	Type	Description
ProcessID	List[int]	The comma separated list of process IDs that could be terminated.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

## DeviceUpdate

This class lists all parameters that were added or updated on the device.

Parameter	Type	Description
UpdateObjectId	integer	The ID of the device to be updated.
ObjectTypeId	string	The unique BCM identifier of a BCM object type.

Parameter	Type	Description
ObjectAttrs	object[ <a href="#">DeviceAttributes</a> ]	This object contains all the parameters that require updating on the device.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DeviceWindowsEvent

This class returns information about a specific Windows event.

Parameter	Type	Description
NtEventType	string	The type of event.
NtEventDateTime	string	The data and time at which the event occurred.
NtEventSource	string	The source of the event.
NtEventCategory	string	The category of the event.
NtEventID	string	The unique ID of the event.
NtEventUser	string	The user who was logged on to the device when the event occurred.
NtEventDescription	string	A description of the event.

### DeviceWindowsEventResult

This class returns the list of events found according to the specified criteria.

Parameter	Type	Description
DeviceWindowsEventResult	LIST[ <a href="#">DeviceWindowsEvent</a> ]	The list of windows events, with each entry providing information about a specific event.
offset	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.
NtEventIndex	integer	The index number that was assigned to the event.
Total	integer	The total number of devices found matching the search query.

Parameter	Type	Description
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### DirectoryInfo

This class contains information for a specific directory.

Parameter	Type	Description
path	string	The complete path of the directory to be created or renamed. Use double backslashes (\\) in the path.
newpath	integer	The complete new path of the directory into which the file is to be renamed. This value is not applicable if the directory is to be created. Use double backslashes (\\) in the path.

### FileInfo

This class contains information for a specific file.

Parameter	Type	Description
path	string	The complete path of the file to be copied or renamed. Use double backslashes (\\) in the path.
newpath	integer	The complete new path to the directory to which the file is to be renamed or copied to. Use double backslashes (\\) in the path.

### FilteredActionsResult

This class returns the list of actions with their availability status.

Parameter	Type	Description
Ping	boolean	true: the action is available, false: the action is not available
ConfigSummary	boolean	true: the action is available, false: the action is not available
Reboot	boolean	true: the action is available, false: the action is not available

Parameter	Type	Description
Shutdown	boolean	true: the action is available, false: the action is not available
WakeUp	boolean	true: the action is available, false: the action is not available
RemoteControl	boolean	true: the action is available, false: the action is not available
FileSystem	boolean	true: the action is available, false: the action is not available
Registry	boolean	true: the action is available, false: the action is not available
Events	boolean	true: the action is available, false: the action is not available
Processes	boolean	true: the action is available, false: the action is not available
Services	boolean	true: the action is available, false: the action is not available
PerformAudit	boolean	true: the action is available, false: the action is not available
FileTransfer	boolean	true: the action is available, false: the action is not available
Values	object[ <a href="#">FilteredActionsValues</a> ]	<p>&lt;p&gt;Translation of the actions:&lt;/p&gt;&lt;p&gt;Key &lt;action name&gt; Value &lt;Translation of the action name&gt;&lt;/p&gt;</p> <p>With the following keys: Ping, Rebbot, Shutdown, WakeUP, RemoteControl, FileSystem, Processes, Registry, Services, Events, ConfigSummary</p>
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### FilteredActionsValues

This class returns the keywords for the returned filtered actions translated to the required language.

Parameter	Type	Description
Ping	string	Ping translated to the target language.
ConfigSummary	string	ConfigSummary translated to the target language.
Reboot	string	Reboot translated to the target language.
Shutdown	string	Shutdown translated to the target language.
WakeUp	string	WakeUp translated to the target language.
RemoteControl	string	RemoteControl translated to the target language.
FileSystem	string	FileSystem translated to the target language.
Registry	string	Registry translated to the target language.
Events	string	Events translated to the target language.
Processes	string	Processes translated to the target language.
Services	string	Services translated to the target language.
PerformAudit	string	PerformAudit translated to the target language.
FileTransfer	string	FileTransfer translated to the target language.

### HardwareKeywordTranslation

This class returns the keywords for the returned hardware inventory translated to the required language.

Parameter	Type	Description
Network Adapter	string	Network Adapter translated to the target language.
DeviceIPAddress	string	DeviceIPAddress translated to the target language.
SerialNumber	string	SerialNumber translated to the target language.
MostUsedUser	string	MostUsedUser translated to the target language.
Type	string	Network Adapter translated to the target language.
DeviceName	string	DeviceName translated to the target language.
LastUpdate	string	LastUpdate translated to the target language.
Domain	string	Domain translated to the target language.

Parameter	Type	Description
Model	string	Model translated to the target language.
OperatingSystemName	string	OperatingSystemName translated to the target language.
TopologyType	string	TopologyType translated to the target language.
Device	string	Device translated to the target language.
DomainName	string	DomainName translated to the target language.
_DB_ATTR_VIMVENDOR_	string	_DB_ATTR_VIMVENDOR_ translated to the target language.
_DB_ATTR_HYPERVISORVENDOR_	string	_DB_ATTR_HYPERVISORVENDOR_ translated to the target language.
AssetUser	string	AssetUser translated to the target language.
AssetAdmin	string	AssetAdmin translated to the target language.

### InventoryInstCreate

This class returns the result when adding an object type to an inventory of a specific device, for example a new object type to the custom inventory. It displays the following information:

Parameter	Type	Description
ObjectId	integer	The unique BCM database identifier of the newly added object type instance.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### JnlpResult

This class returns the XML content of the JNLP file.

Parameter	Type	Description
JNLPFile	string	The name and path to the JNLP file.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### KeywordTranslation

This class returns the keywords for the financial summary translated to the required language.



Parameter	Type	Description
PurchaseCostofAsset	string	PurchaseCostofAsset translated to the target language.
ServiceStartDate	string	ServiceStartDate translated to the target language.
SupportProvider	string	SupportProvider translated to the target language.
SupportExpirationDate	string	SupportExpirationDate translated to the target language.
WarrantyExpirationDate	string	WarrantyExpirationDate translated to the target language.
ResidualValue	string	ResidualValue translated to the target language.
ActualCost	string	ActualCost translated to the target language.

### OpRuleHistoryResult

This class returns the list of operational rules assigned to a specific device.

Parameter	Type	Description
Result	string	<p>A collection of hash entries with the following keys:</p> <ul style="list-style-type: none"> <li>• ObjectId: The Id of the operational rule.</li> <li>• ObjectName: The name of the operational rule.</li> <li>• Status: The translated status of the rule deployment.</li> <li>• ErrorText: The translated error text.</li> <li>• Total Integer total number of available values</li> </ul>
Total	integer	The number of operational rules returned.
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

### SessionOperationResult

This class returns the result for an open session operation.

Parameter	Type	Description
SessionOperationResult	object[ <a href="#">SessionResult</a> ]	The class returning the session identifier.

Parameter	Type	Description
ErrorCode	integer	The error code of the executed operation, which can be either 0 for a successful operation or 1 for a failed one.

**SessionResult**

This class returns the identifier of a newly opened session.

Parameter	Type	Description
id	integer	The unique BCM database identifier.

**SoftwareKeywordTranslation**

This class returns the keywords for the returned software inventory translated to the required language.

Parameter	Type	Description
Name	string	Name translated to the target language.
Version	string	Version translated to the target language.
Software Manufacturer	string	Software Manufacturer translated to the target language.
SupportExpirationDate	string	SupportExpirationDate translated to the target language.
Type	string	Type translated to the target language.
Installation Directory	string	Installation Directory translated to the target language.

**Group**

The group object type allows you to execute operations on specific groups as well as some folder objects: device groups, user groups and package folders.

[Group Operations](#)

[Return Classes](#)

**Group Operations**

The following operations are available for the group object:

**GET**

The following get operations are available:

Action	Description
<a href="#">/group/{groupid}/hierarchy</a>	Get device groups hierarchy.
<a href="#">/group/subgroups</a>	Get device groups under root node.
<a href="#">/group/subgroup/{groupid}/subgroups</a>	Get device group subgroups.
<a href="#">/group/devices</a>	Get devices under root node.
<a href="#">/group/subgroup/{groupid}/devices</a>	Get devices of group.
<a href="#">/user/subgroups</a>	Get user groups under root node.
<a href="#">/user/group/{groupid}/subgroups</a>	Get user group subgroups.
<a href="#">/user/users</a>	Get users under root node.
<a href="#">/user/group/{groupid}/users</a>	Get users of group.
<a href="#">/package/subfolders</a>	Get package folders under root node.
<a href="#">/package/folder/{folderId}/subfolders</a>	Get package folder subfolders.
<a href="#">/package/packages</a>	Get packages under root node.
<a href="#">/package/folder/{folderId}/packages</a>	Returns the packages of a specific package folder.

## POST

The following post operations are available:

Action	Description
<a href="#">/group/{groupIds}/wakeup</a>	Wake a group of devices up.

## GET Operations

### [/group/{groupid}/hierarchy](#)

This operation returns the device groups hierarchy of a specific group.

### Syntax

```
GET /group/{groupId}/hierarchy?
offset=OffsetStart&count=EntriesPerPage&language=OutputLanguage
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the device group. This value is returned by the <code>GET /group/subgroups</code> , <code>GET /group/{groupId}/hierarchy</code> and <code>GET /group/subgroup/{groupId}/subgroups</code> operations.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.

## Returns

Returns the class [GroupValuesInfo](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all groups below the group with the ID is 195 and displays its results in Japanese.

### Request URL

<https://localhost:1611/api/1/group/195/hierarchy?offset=0&count=10&language=Japanese>

### Response Body

```
{
  "Values": [
    {
      "ObjectId": 197,
      "ObjectName": "Client Devices",
      "ObjectTypeName": "_DB_OBJECTTYPE_DEVICEGROUP_",
      "ObjectSmallIcon": "devicegrp/group_dynamic.png"
    },
    {
      "ObjectId": 199,
      "ObjectName": "Lost devices - 30 days",
      "ObjectTypeName": "_DB_OBJECTTYPE_DEVICEGROUP_",
      "ObjectSmallIcon": "devicegrp/group_dynamic.png"
    },
    {
      "ObjectId": 196,
      "ObjectName": "#####",
      "ObjectTypeName": "_DB_OBJECTTYPE_DEVICEGROUP_",
      "ObjectSmallIcon": "devicegrp/group_dynamic.png"
    },
    {
      "ObjectId": 198,
      "ObjectName": "#####",
      "ObjectTypeName": "_DB_OBJECTTYPE_DEVICEGROUP_",
      "ObjectSmallIcon": "devicegrp/group_dynamic.png"
    }
  ]
}
```

```
    }
  ],
  "Total": 4,
  "ErrorCode": 0
}
```

The group in this case has three groups and one individual device therefore the total of 4 children.

**/group/subgroups**

This operation returns the device groups located directly at the root level of the main device groups node.

**Syntax**

GET /group/subgroups?  
brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowHierarchy&offset=OffsetStart&count=Entri

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the device groups or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the group ( <i>name</i> ), the date at which the group was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).
hierarchy	query	no	boolean	Select from this drop-down box if the devices are listed in the return class as well, ( <i>true</i> ) for include the list, ( <i>false</i> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

**Returns**

Returns the class *GroupDevices*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all subgroups of the main **Device Groups** top node, providing all details and with pagination activated to 10 per page and starting with the first entry. The devices under the node will not be shown.

### Request URL

```
https://localhost:1611/api/1/group/subgroups?
brief=false&sort=name&asc=true&hierarchy=false&offset=1&count=10
```

### Response Body

```
{
  "groups": [
    {
      "id": 138,
      "name": "All Devices",
      "notes": "",
      "createDate": 1361527321,
      "updateDate": 1379929769
    },
    {
      "id": 255,
      "name": "All my devices",
      "notes": "",
      "createDate": 1379929820,
      "updateDate": 1379929827
    },
    {
      "id": 183,
      "name": "New Device Group",
      "notes": "",
      "createDate": 1365501351
    }
  ],
}
```

```
{
  {
    "id": 195,
    "name": "Out-of-the-Box#####",
    "notes": "",
    "createDate": 1373360810,
    "updateDate": 1373360815
  }
},
"devices": []
}
```

The top node in this case has two groups.

### **/group/subgroup/{groupId}/subgroups**

This operation returns the device groups that are members of a specific other device group.

### **Syntax**

GET /group/subgroup/{groupId}/subgroups?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowHierarchy&offset=OffsetStart&count=Entri

### **Parameters**

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the group. This value is returned by the <code>GET /group/subgroups</code> , <code>GET /group/{groupId}/hierarchy</code> and <code>GET /group/subgroup/{groupId}/subgroups</code> operations.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the device groups or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the group ( <code>name</code> ), the date at which the group was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if the devices are listed in the return class as well, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include. If this option is turned off, this will returns the same result as the <code>GET /group/{groupId}/hierarchy</code> operation.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *GroupDevices*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all children (subgroups and devices) of the group with the ID 198.

### Request URL

```
https://localhost:1611/api/1/group/subgroup/195/subgroups?
brief=false&sort=name&asc=true&hierarchy=true&offset=1&count=10
```

## Response Body

```
{
  "groups": [],
  "devices": [
    {
      "id": 1001,
      "name": "NewTestDevice",
      "notes": "",
      "createDate": 1367240626,
      "updateDate": 1367243448
    }
  ]
}
```

## /group/devices

This operation returns the devices located directly at the root level of the main device groups node.

## Syntax

GET /group/devices?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the devices or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the device ( <i>name</i> ), the date at which the device was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).
hierarchy	query	no	boolean	Select from this drop-down box if the devices are listed in the return class, ( <i>true</i> ) for include the list, ( <i>false</i> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

Returns

Returns the class *Devices*.

Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

Example

This example finds all devices directly below the **Device Groups** top node.

Request URL

https://localhost:1611/api/1/group/devices?brief=false&sort=name&asc=true&count=100

Response Body

```
{
  "devices": []
}
```

There is no device located directly under the top node.

/group/subgroup/{groupId}/devices

This operation returns the devices that are members of a specific device group.

Syntax

GET /group/subgroup/{groupId}/devices?  
brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the group. This value is returned by the <code>GET /group/subgroups</code> , <code>GET /group/{groupId}/hierarchy</code> and <code>GET /group/subgroup/{groupId}/subgroups</code> operations.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the devices or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the device ( <code>name</code> ), the date at which the device was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).

Name	Type	Mand.	Data Type	Description
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the devices are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [Devices](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all member devices of a device group with ID 255.

**Request URL**

```
https://localhost:1611/api/1/group/subgroup/255/devices?
brief=false&sort=name&asc=true&offset=1&count=10
```

**Response Body**

```
{
  "devices": [
    {
      "id": 10001,
      "name": "MyMaster",
      "notes": "",
      "createDate": 1467240626,
      "updateDate": 1567243448
    }
    {
      "id": 1001,
      "name": "NewTestDevice",
      "notes": "",
      "createDate": 1367240626,
      "updateDate": 1367243448
    }
  ]
}
```

**/user/subgroups**

This operation returns the user groups located directly at the root level of the main user groups node.

**Syntax**

GET /user/subgroups?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowUsers&offset=OffsetStart&count=EntriesPer

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the user groups or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the group ( <i>name</i> ), the date at which the group was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).

Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if the users are listed in the return class as well, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class `GroupUsers`.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all groups directly below the **User Groups** top node.

### Request URL

```
https://localhost:1611/api/1/group/100/hierarchy?count=100&language=English
```

### Response Body

```
{
```



```

"groups": [
  {
    "id": 178,
    "name": "All Users",
    "notes": "",
    "createDate": 1361527328,
    "updateDate": 1361527357
  },
  {
    "id": 242,
    "name": "Out-of-the-Box#####",
    "notes": "",
    "createDate": 1373360814,
    "updateDate": 1373360837
  }
]
}

```

The top node in this case has two groups.

### **/user/group/{groupId}/subgroups**

This operation returns the user groups that are members of a specific other user group.

### **Syntax**

GET /user/group/{groupId}/subgroups?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowUsers&offset=OffsetStart&count=EntriesPer

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the user group. User group IDs are returned by the <code>GET /user/subgroups</code> or <code>GET /user/group/{groupId}/subgroups</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the user groups or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the group ( <code>name</code> ), the date at which the group was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if the users are listed in the return class as well, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *GroupDevices*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all children, that is, all subgroups and users, of a group with ID 242.

### Request URL

```
https://localhost:1611/api/1/user/group/242/subgroups?
brief=false&sort=name&asc=true&hierarchy=true&offset=0&count=10
```

### Response Body

```
{
  "groups": [
    {
      "id": 244,
      "name": "Inactive Users",
      "notes": "All imported users that are disabled in the directory server",
      "createDate": 1373360814,
      "updateDate": 1373360837
    },
    {
      "id": 245,
      "name": "Users Imported from directory",
      "notes": "",
      "createDate": 1373360814,
      "updateDate": 1373360837
    },
    {
      "id": 246,
      "name": "Users with advertised rules",
      "notes": "",
      "createDate": 1373360814,
      "updateDate": 1373360837
    },
    {
      "id": 243,
      "name": "#####",
      "notes": "",
      "createDate": 1373360814,
      "updateDate": 1373360837
    }
  ],
  "users": [
    {
      "id": 1001,
      "name": "New User",
      "notes": "",
      "domain": "",
      "login": "user",
      "isEnabled": true,
      "createDate": 1379937481,
      "updateDate": 1379937498
    }
  ]
}
```

## /user/users

This operation returns the users located directly at the root level of the main user groups node.

## Syntax

GET /user/users?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the users or also other available information, true for IDs only, false for IDs plus information such as name, creation date, and so on.

Name	Type	Mand.	Data Type	Description
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the group ( <code>name</code> ), the date at which the group was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the user groups are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [Users](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all users directly below the **User Groups** top node.

### Request URL

```
https://localhost:1611/api/1/user/users?brief=false&sort=name&asc=true&offset=0&count=10
```

### Response Body

```
{
  "users": [
    {
      "id": 1002,
      "name": "SysAdmin",
      "notes": "",
      "domain": "/myspycompany/hq",
      "login": "sysadmin",
      "isEnabled": true,
      "createDate": 1379938705,
      "updateDate": 1379938801
    }
  ]
}
```

## /user/group/{groupId}/users

This operation returns the user that are members of a specific user group.

### Syntax

GET /user/group/{groupId}/users?

brief>ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowUsers&offset=OffsetStart&count=EntriesPer

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the user group. User group IDs are returned by the <code>GET /user/subgroups</code> or <code>GET /user/group/{groupId}/subgroups</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the users or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the group ( <i>name</i> ), the date at which the group was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).
hierarchy	query	no	boolean	Select from this drop-down box if the device groups are listed in the return class, ( <i>true</i> ) for include the list, ( <i>false</i> ) for don't include.

Name	Type	Mand.	Data Type	Description
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *Users*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all member users of a group with the ID 243.

### Request URL

```
https://localhost:1611/api/1/user/group/243/users?brief=false&sort=name&asc=true&count=100
```

### Response Body

```
{
  "users": [
    {
      "id": 1000,
      "name": "MySpyCompany\\jblond",
      "notes": "",
    }
  ]
}
```



```

    "domain": "MySpyCompany",
    "login": "jblond",
    "isEnabled": true,
    "createDate": 1361527434
  }
]
}

```

## /package/subfolders

This operation returns the package folders located directly at the root level of the main packages node.

## Syntax

GET /package/subfolders?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowPkgs&offset=OffsetStart&count=EntriesPer

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the package folders or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the packages are listed in the return class as well, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [GroupPackages](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all package folders and packages directly below the **Packages** top node.

### Request URL

```
https://localhost:1611/api/1/package/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&offset=1&count=10
```

### Response Body

```
{
  "folders": [],
  "packages": [
    {
      "id": 1007,
      "name": "DiagFiles.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1375352384,
      "updateDate": 1375352384
    },
    {
      "id": 1001,
      "name": "New Package.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1366033698,
      "updateDate": 1366033698
    },
    {
      "id": 1000,
      "name": "NmapFiles.cst",
      "notes": "",
      "type": "CUSTOM",

```

```

    "createDate": 1361527376,
    "updateDate": 1361527376
  }
}

```

In this example there are currently three packages but no subfolder under the root node.

### **/package/folder/{folderId}/subfolders**

This operation returns the package folders that are members of a specific package folder.

### **Syntax**

GET /package/folder/{folderId}/subfolders?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowPkgs&offset=OffsetStart&count=EntriesPer

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the package folder. This value is returned by either the GET /package/subfolders or a GET /package/folder/{folderId}/subfolders operation.

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the package folders or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the packages are listed in the return class as well, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [GroupPackages](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.

Error Code	Description
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all groups directly below the **Device Groups** top node, whose ID is 100.

### Request URL

`https://localhost:1611/api/1/group/100/hierarchy?count=100&language=English`

### Response Body

```
{
  "Values": [
    {
      "ObjectId": 138,
      "ObjectName": "All Devices",
      "ObjectTypeName": "_DB_OBJECTTYPE_DEVICEGROUP_",
      "ObjectSmallIcon": "devicegrp/group_dynamic.png"
    },
    {
      "ObjectId": 183,
      "ObjectName": "New Device Group",
      "ObjectTypeName": "_DB_OBJECTTYPE_DEVICEGROUP_",
      "ObjectSmallIcon": "devicegrp/group.png"
    }
  ],
  "Total": 2,
  "ErrorCode": 0
}
```

The top node in this case has two groups.

## /package/packages

This operation returns the packages located directly at the root level of the main packages node.

### Syntax

GET /package/packages?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the packages or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the package ( <code>name</code> ), the date at which the package was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [Packages](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.

Error Code	Description
16010	Login failed.

## Example

This example finds all packages that are stored directly below the main **Packages** top node, providing all details and with pagination activated to 10 per page and starting with the first entry. The devices under the node will not be shown.

### Request URL

```
https://localhost:1611/api/1/package/packages?brief=false&sort=name&asc=true&offset=1&count=10
```

### Response Body

```
{
  "packages": [
    {
      "id": 1007,
      "name": "DiagFiles.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1375352384,
      "updateDate": 1375352384
    },
    {
      "id": 1001,
      "name": "New Package.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1366033698,
      "updateDate": 1366033698
    },
    {
      "id": 1000,
      "name": "NmapFiles.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1361527376,
      "updateDate": 1361527376
    }
  ]
}
```

## /package/folder/{folderId}/packages

This operation returns the packages that are members of a specific package folder.

### Syntax

```
GET /package/folder/{folderId}/packages?
```

```
brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the package folder. This value is returned by either the <code>GET /package/subfolders</code> or a <code>GET /package/folder/{folderId}/subfolders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the packages or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the package ( <i>name</i> ), the date at which the package was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.



Name	Type	Mand.	Data Type	Description
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *Packages*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds all groups directly below the **Device Groups** top node, whose ID is 100.

### Request URL

```
https://localhost:1611/api/1/group/100/hierarchy?count=100&language=English
```

### Response Body

```
{
  "Values": [
    {
      "ObjectId": 138,
      "ObjectName": "All Devices",
      "ObjectTypeName": "_DB_OBJECTTYPE_DEVICEGROUP_",
      "ObjectSmallIcon": "devicegrp/group_dynamic.png"
    },
    {
      "ObjectId": 183,
      "ObjectName": "New Device Group",

```

```
    "ObjectName": "_DB_OBJECTTYPE_DEVICEGROUP_",
    "ObjectSmallIcon": "devicegrp/group.png"
  }
],
"Total": 2,
"ErrorCode": 0
}
```

The top node in this case has two groups.

POST Operations

/group/{groupIds}/wakeup

This operation wakes a specific group of devices up for further operation execution.

Syntax

```
POST /group/{groupIds}/wakeup?retrycount=#Retries&retryinterval=Interval
```

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
groupIds	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM identifier of the group or groups to wake up. If more than one group is listed, the groups must be separated by a colon (,). These are returned by the GET /group/subgroups , GET /group/{groupId}/hierarchy and GET /group/subgroup/{groupId}/subgroups operations.

Name	Type	Mand.	Data Type	Description
retrycount	query	no	integer	The number of retries to wake up the member of the group if the preceding tentative failed.
retryinterval	query	no	integer	The interval between two wakeup retries in seconds.

## Returns

Returns the class *GroupOperationResult*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example tries to wake up all devices of two groups, that were found via the `GET /group/{groupId}/hierarchy` operation.

### Request URL

```
https://localhost:1611/api/1/group/138%2C183/wakeup?retrycount=1&retryinterval=1000
```

### Response Body

```
{
  "fault": {
    "code": 1,
```

```

    "uri": "/api/1/group/138,183/wakeup"
  }
}

```

The operation is only executed once and it failed, that is at least one of the devices of the groups could not be woken up.

## Group Return Classes

The following classes can be returned by the different group operations:

Class	Description
<i>DeviceGroups</i>	The details of a specific device group.
<i>Devices</i>	Search for devices in the BCM database.
<i>GroupDevices</i>	Get devices under root node.
<i>GroupOperationResult</i>	Wakeup operation result on group members.
<i>GroupPackages</i>	This class returns the list of all package folders, either located directly under the main packages node or under a specific folder, as well as all member packages if specified.
<i>GroupUsers</i>	Returns the list of all user groups and all users if specified.
<i>GroupValues</i>	Return list of device group values.
<i>GroupValuesInfo</i>	Return all children of a device group.
<i>Packages</i>	This class returns the list of all packages, either located directly under the main packages node or under a specific folder.
<i>PackagesFolders</i>	This class returns the available information about one specific package folder.
<i>UserGroups</i>	Details of a user group member of a specific group.
<i>Users</i>	Returns the list of all users.

## DeviceGroups

This class returns the available information about one specific device group.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the group.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.

Parameter	Type	Description
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

## Devices

This operation finds specific devices in the BCM database.

Parameter	Type	Description
Id	integer	The unique BCM database ID of the device.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

## GroupDevices

This class returns all devices and device groups located directly at the root level of the main device groups node.

Parameter	Type	Description
devicesgroups	object[ <a href="#">DeviceGroups</a> ]	The list of all members found under the requested group. One entry per member which is detailed via a <code>DeviceGroups</code> object each.
devices	object[ <a href="#">Devices</a> ]	The list of all members found under the requested group. One entry per member which is detailed via a <code>DeviceGroups</code> object each.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### GroupOperationResult

This class returns the result of a wake up operation on the members of one or more device groups.

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### GroupPackages

This class returns the list of all package folders, either located directly under the main packages node or under a specific folder, as well as all member packages if specified.

Parameter	Type	Description
folders	object[ <a href="#">PackagesFolders</a> ]	This class returns the available information about one specific package folder.
packages	object[ <a href="#">Packages</a> ]	The list of all members found under the requested folder. One entry per member which is detailed via a <code>Packages</code> object each.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### GroupUsers

This class returns the list of all user groups, either located directly under the main user groups node or under a specific group, as well as all member users if specified.

Parameter	Type	Description
usergroups	object[ <a href="#">UserGroups</a> ]	The list of all members found under the requested group. One entry per member which is detailed via a <code>UserGroups</code> object each.
users	object[ <a href="#">Users</a> ]	The list of all members found under the requested group. One entry per member which is detailed via a <code>Users</code> object each.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### GroupValues

This class returns that attributes and their values of a specific device group.

Parameter	Type	Description
objectId	integer	The unique BCM database identifier of the object.
ObjectName	string	The name of the object type.
ObjectTypeName	string	The internal BCM database name under which this object type is known, for example <code>_DB_OBJECTTYPE_DEVICEGROUP_</code> for a device group.
ObjectSmallIcon	string	The name and relative path of the icon representing the current state of the found object.

### GroupValuesInfo

This class returns the list of all groups and devices that were found under the requested device group.

Parameter	Type	Description
Values	object[ <a href="#">GroupValues</a> ]	The list of all members found under the requested group. One entry per member which is detailed via a <code>GroupValues</code> object each.
Total	integer	The total number of members found.
ErrorCode	integer	The error code returned by the operation if the operation failed or <code>0</code> for a successful operation.

### Packages

This class returns the list of all packages, either located directly under the main packages node or under a specific folder.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the package.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
type	string	The type of the package. <i>Custom</i> indicates that the package may either be a manually created custom package or a patch package.
createDate	string	The date and time at which the object was created in the BCM database.

Parameter	Type	Description
updateDate	string	The date and time at which the object was last modified in the BCM database.

## PackagesFolders

This class returns the available information about one specific package folder.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the package folder.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

## UserGroups

This class returns the details that are available for a user group that is a member of a specific group.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the user.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

## Users

This class returns the list of all users, either located directly under the main user groups node or under a specific group.



Parameter	Type	Description
Id	integer	The unique BCM database identifier of the user.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
domain	string	The name of the domain for which the user is defined.
logon	string	The login name of the user with which he can log on to any client in the network.
isEnabled	boolean	Indicates if the user and his login are currently enabled, that is, if when using this login on a client he can access the specified user options.
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

## Operational Rule

The operational rule object type allows you to execute operations on specific rules.

[Operational rule operations](#)

[Return Classes](#)

## Operational Rule Operations

The following operations are available for the operational rule object:

### GET

The following get operations are available:

Action	Description
<a href="#">/oprul/subfolders</a>	Get the list of root subfolders.
<a href="#">/oprul/folder/{folderId}/subfolders</a>	Get list of operational rules subfolders.
<a href="#">/oprul/folder/{folderId}</a>	Get operational rules folder details.

Action	Description
<a href="#">/oprule/rules</a>	Get list of root operational rules.
<a href="#">/oprule/rule/{ruleId}</a>	Get operational rules list of specific folder.
<a href="#">/oprule/folder/{folderId}/rules</a>	Get details of operational rule.
<a href="#">/oprule/rule/{ruleId}/step/assignments</a>	Get operational rule step list.
<a href="#">/oprule/rule/{ruleId}/step/assignment/{assignmentId}</a>	Get operational rule step assignment details.
<a href="#">/oprule/icons</a>	Get list of available MyApps icons.
<a href="#">/oprule/step/classes</a>	Get operational rule step class list.
<a href="#">/oprule/steps</a>	Get operational rule step list.
<a href="#">/oprule/step/{stepTypeId}</a>	Get step parameter list.
<a href="#">/oprule/rule/{ruleId}/device/assignments</a>	Get operational rule device assignment details list.
<a href="#">/oprule/rule/device/assignment/{assignmentId}</a>	Get operational rule device assignment details.
<a href="#">/oprule/rule/{ruleId}/device/group/assignments</a>	Get operational rule device group assignment details list.
<a href="#">/oprule/rule/device/group/assignment/{assignmentId}</a>	Get operational rule device group assignment details.
<a href="#">/oprule/rule/{ruleId}/user/assignments</a>	This operation returns the list of all users that are assigned to a specific operational rule.
<a href="#">/oprule/rule/user/assignment/{assignmentId}</a>	Get operational rule user assignment details.
<a href="#">/oprule/rule/{ruleId}/user/group/assignments</a>	Get operational rule user group assignment details list.
<a href="#">/oprule/rule/user/group/assignment/{assignmentId}</a>	Get operational rule user group assignment details.
<a href="#">/oprule/rule/{ruleId}/dependencies</a>	Get dependent operational rules list.
<a href="#">/oprule/rule/{ruleId}/packages</a>	Get list of packages assigned to operational rule.
<a href="#">/schedule/{scheduleId}</a>	Get schedule details.

## PUT

The following get operations are available:

Action	Description
<a href="#">/oprule/subfolder</a>	Create operational rule folder under root.
<a href="#">/oprule/folder/{folderId}/subfolder</a>	Create operational rules subfolder under specific folder.

Action	Description
<a href="#">/oprule/rule</a>	Create operational rule under root.
<a href="#">/oprule/folder/{folderId}/rule</a>	Get operational rules list of specific folder.
<a href="#">/oprule/folder/{folderId}</a>	Update operational rules folder parameters.
<a href="#">/oprule/rule/{ruleId}</a>	Update operational rule parameters.
<a href="#">/oprule/rule/{ruleId}/step/{stepTypeId}</a>	Assign operational rule step.
<a href="#">/oprule/rule/{ruleId}/device/{deviceId}</a>	Assign device to operational rule.
<a href="#">/oprule/rule/{ruleId}/device/group/{groupId}</a>	Assign device group to operational rule.
<a href="#">/oprule/rule/{ruleId}/user/group/{groupId}</a>	Assign user group to operational rule.
<a href="#">/oprule/rule/{ruleId}/user/{userId}</a>	Assign user to operational rule.
<a href="#">/oprule/rule/{ruleId}/step/assignment/{assignmentId}</a>	Update step assignment details.
<a href="#">/oprule/rule/device/assignment/{assignmentId}</a>	Update operational rule device assignment parameters.
<a href="#">/oprule/rule/device/group/assignment/{assignmentId}</a>	Update operational rule device group assignment parameters.
<a href="#">/oprule/rule/user/assignment/{assignmentId}</a>	Update operational rule user assignment parameters.
<a href="#">/oprule/rule/user/group/assignment/{assignmentId}</a>	Update operational rule user group assignment parameters.
<a href="#">/oprule/rule/{ruleId}/dependency/{dependentRuleId}</a>	Assign operational rule dependency.
<a href="#">/oprule/rule/{ruleId}/package/{packageId}</a>	Add operational rule package.
<a href="#">/schedule/{scheduleId}</a>	Update schedule details.

## DELETE

The following get operations are available:

Action	Description
<a href="#">/oprule/folder/{folderId}</a>	Delete operational rules folder.
<a href="#">/oprule/rule/{ruleId}/step/assignment/{assignmentId}</a>	Delete operational rule step assignment.
<a href="#">/oprule/rule/{ruleId}</a>	Delete operational rule.
<a href="#">/oprule/rule/device/assignment/{assignmentId}</a>	Delete operational rule device assignment.
<a href="#">/oprule/rule/device/group/assignment/{assignmentId}</a>	Delete operational rule device group assignment.

Action	Description
<a href="#">/oprule/rule/user/assignment/{assignmentId}</a>	Delete operational rule user assignment.
<a href="#">/oprule/rule/user/group/assignment/{assignmentId}</a>	Delete operational rule user group assignment.
<a href="#">/oprule/rule/{ruleId}/dependency/{dependentRuleId}</a>	Delete operational rule user group assignment.
<a href="#">/oprule/rule/{ruleId}/package/{packageId}</a>	Unassign operational rule package.

## GET Operations

### /oprule/subfolders

This operation returns the list of operational rule folders located directly below the top node.

### Syntax

GET /oprule/subfolders?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowRules&offset=OffsetStart&count=EntriesPerPage

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending <code>true</code> or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the operational rules contained in the respective folder are listed in the return class, <code>true</code> for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.

Name	Type	Mand.	Data Type	Description
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *OperationalRuleFolderArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

The following subfolders are listed with further information and their hierarchy after only the out-of-the-box objects are imported and no other folders or rules are created yet.

### Request URL

```
https://localhost:1611/api/1/oprule/subfolders?
brief=false&sort=name&asc=true&hierarchy=false&count=100
```

### Response Body

```
{
  "total": 6,
  "folders": [
    {
      "id": 169,
```

```

    "name": "Compliance Data Collection",
    "notes": "",
    "createDate": 1361527322,
    "updateDate": 1361527338
  },
  {
    "id": 251,
    "name": "<term conref = \"../variables/variables.xml#id10B8A000W5Z/term_FullName\"/>
Upgrade",
    "notes": "",
    "createDate": 1375281716,
    "updateDate": 1375281739
  },
  {
    "id": 171,
    "name": "Inventory Management",
    "notes": "",
    "createDate": 1361527322,
    "updateDate": 1361527338
  },
  {
    "id": 230,
    "name": "Out of the Box",
    "notes": "",
    "createDate": 1373360811,
    "updateDate": 1373360822
  },
  {
    "id": 172,
    "name": "Windows Peripheral Device Management",
    "notes": "",
    "createDate": 1361527322,
    "updateDate": 1361527338
  },
  {
    "id": 170,
    "name": "Windows Power Management",
    "notes": "",
    "createDate": 1361527322,
    "updateDate": 1361527338
  }
]
}

```

### **/oprule/folder/{folderId}/subfolders**

This operation returns the list of subfolders of a specific operational rules folder.

### **Syntax**

GET /oprule/folder/{folderId}/subfolders?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowRules&offset=OffsetStart&count=EntriesPer

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rules subfolder. This value is returned by either the <i>/oprule/subfolders</i> or another <i>/oprule/folder/{folderId}/subfolders</i> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <i>name</i> ), the date at which the folder was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending <i>true</i> or descending ( <i>false</i> ).

Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if the operational rules contained in the respective folder are listed in the return class, <code>true</code> for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [OperationalRuleFolderArray](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This operation returns all children, folders and rules of an operational rules folder with the ID 169.



**Request URL**

```
https://localhost:1611/api/1/oprule/folder/169/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&count=100
```

**Response Body**

```
{
  "total": 3,
  "folders": [],
  "rules": [
    {
      "id": 1001,
      "name": "Power Savings Inventory",
      "notes": "",
      "createDate": 1361527322,
      "updateDate": 1361527339
    },
    {
      "id": 1002,
      "name": "Security Inventory (Advanced)",
      "notes": "",
      "createDate": 1361527322,
      "updateDate": 1361527340
    },
    {
      "id": 1003,
      "name": "Security Inventory (Basic)",
      "notes": "",
      "createDate": 1361527322,
      "updateDate": 1361527340
    }
  ]
}
```

**/oprule/folder/{folderId}**

This operation returns the details of a specific operational rules folder.

**Syntax**

```
GET /oprule/folder/{folderId}
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rules subfolder. This value is returned by either the <i>/oprule/subfolders</i> or another <i>/oprule/folder/{folderId}/subfolders</i> operation.

## Returns

Returns the class *OperationalRuleFolderValues*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

### Example

This operation returns the detail information of an operational rules folder with the ID 169.

#### Request URL

```
https://localhost:1611/api/1/oprule/folder/169
```

#### Response Body

### /oprule/rules

This operation returns the list of operational rules directly located under the Operational Rules top node.

### Syntax

```
GET /oprule/rules?
```

```
brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the operational rules or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.

Name	Type	Mand.	Data Type	Description
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the operational rules ( <code>name</code> ), the type of the operational rules ( <code>type</code> ), the URL contained in the operational rules (Quick Link rules only - <code>url</code> ), the date at which the operational rules was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending <code>true</code> or descending ( <code>false</code> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [OperationalRuleArray](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.

Error Code	Description
16010	Login failed.

## Example

This operation returns the list of operational rules that are located directly under the main **Operational Rules** node.

### Request URL

<https://localhost:1611/api/1/oprule/rules?brief=false&sort=name&asc=true&count=100>

### Response Body

```
{
  "total": 7,
  "rules": [
    {
      "id": 1066,
      "name": "Activate Peripheral logging",
      "type": "Operational Rule",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/Kiosk/oprule.png",
      "notes": "",
      "createDate": 1377005691,
      "updateDate": 1377006681
    },
    {
      "id": 1064,
      "name": "DiagFiles.cst",
      "type": "Software Distribution",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/kiosk/oprules_pkg.png",
      "notes": "",
      "createDate": 1375352385,
      "updateDate": 1375352385
    },
    {
      "id": 1030,
      "name": "File Share",
      "type": "Quick Link",
      "url": "http://www.google.com",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/Kiosk/oprule.png",
      "notes": "",
      "createDate": 1363355598,
      "updateDate": 1363355621
    },
    {
      "id": 1029,
      "name": "Google",
      "type": "Quick Link",
      "url": "http://www.google.com",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/Kiosk/oprule.png",
      "notes": "",
      "createDate": 1363352924,
      "updateDate": 1363352970
    },
    {
      "id": 1032,
      "name": "New Package.cst",
      "type": "Software Distribution",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/kiosk/oprules_pkg.png",
      "notes": ""
    }
  ]
}
```

```

    "createDate": 1366033702,
    "updateDate": 1366033702
  },
  {
    "id": 1027,
    "name": "NmapFiles.cst",
    "type": "Software Distribution",
    "deploymentFromExternalIntegration": "Not available in integrated solutions",
    "kioskIcon": "or/kiosk/oprules_pkg.png",
    "notes": "",
    "createDate": 1361527500,
    "updateDate": 1361527500
  },
  {
    "id": 1031,
    "name": "Update Antivirus",
    "type": "Quick Link",
    "url": "www.google.com",
    "deploymentFromExternalIntegration": "Not available in integrated solutions",
    "kioskIcon": "or/Kiosk/oprule.png",
    "notes": "Please update the antivirus software on your device. Click the provided link to start the update process.",
    "createDate": 1363355638,
    "updateDate": 1363356285
  }
]
}

```

### **/oprule/rule/{ruleId}**

This operation returns the details of an operational rule.

### **Syntax**

GET /oprule/rule/{ruleId}

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

## Returns

Returns the class [OperationalRuleValues](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

## Example

This example returns the details of an operational rule with ID 1066.

### Request URL

```
https://localhost:1611/api/1/oprule/rule/1066
```

### Response Body

```
{
  "rule": {
    "id": 1066,
    "name": "Activate Peripheral logging",
    "type": "Operational Rule",
    "deploymentFromExternalIntegration": "Not available in integrated solutions",
    "kioskIcon": "or/Kiosk/oprule.png",
    "notes": "",
    "createDate": 1377005691,
    "updateDate": 1377006681
  }
}
```

## /oprule/folder/{folderId}/rules

This operation returns the list of operational rules of a specific folder.

## Syntax

```
GET /oprule/folder/{folderId}/rules?
```

```
brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage
```

## Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rules folder. This value is returned by either the <i>/oprule/subfolders</i> or another <i>/oprule/folder/{folderId}/subfolders</i> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the operational rules or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the operational rules ( <i>name</i> ), the type of the operational rules ( <i>type</i> ), the URL contained in the operational rules (Quick Link rules only - <i>url</i> ), the date at which the operational rules was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending <i>true</i> or descending ( <i>false</i> ).

Name	Type	Mand.	Data Type	Description
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *OperationalRuleArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the list of operational rules that are members of a folder with ID 169.

### Request URL

```
https://localhost:1611/api/1/oprule/folder/169/rules?brief=false&sort=name&asc=true&count=100
```

### Response Body

```
{
  "total": 3,
```

```

"rules": [
  {
    "id": 1001,
    "name": "Power Savings Inventory",
    "type": "Operational Rule",
    "deploymentFromExternalIntegration": "Not available in integrated solutions",
    "kioskIcon": "or/Kiosk/oprule.png",
    "notes": "",
    "createDate": 1361527322,
    "updateDate": 1361527339
  },
  {
    "id": 1002,
    "name": "Security Inventory (Advanced)",
    "type": "Operational Rule",
    "deploymentFromExternalIntegration": "Not available in integrated solutions",
    "kioskIcon": "or/Kiosk/oprule.png",
    "notes": "",
    "createDate": 1361527322,
    "updateDate": 1361527340
  },
  {
    "id": 1003,
    "name": "Security Inventory (Basic)",
    "type": "Operational Rule",
    "deploymentFromExternalIntegration": "Not available in integrated solutions",
    "kioskIcon": "or/Kiosk/oprule.png",
    "notes": "",
    "createDate": 1361527322,
    "updateDate": 1361527340
  }
]
}

```

### **/oprule/rule/{ruleId}/step/assignments**

This operation returns the list of steps that are assigned to a specific operational rule.

### **Syntax**

GET /oprule/rule/{ruleId}/step/assignments?brief=ShowDetails

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the operational rule step assignments or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, type, and so on.

## Returns

Returns the class *OperationalRuleValues*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example provides the list of steps of the operational rule with ID 1066 including their details, that is, the list of each step parameter and their values

#### Request URL

<https://localhost:1611/api/1/oprule/rule/1066/step/assignments?brief=true>

#### Response Body

```
{
  "assignments": [
    {
      "id": 1272,
      "position": 1,
      "step": {
        "id": 1087,
        "params": [
          {
            "id": 1453,
            "label": "_DB_STEPPARAM_MODULENAMES_",
            "name": "ModuleName",
            "type": "Enum",
            "value": "DeviceManagement"
          },
          {
            "id": 1454,
            "label": "_DB_STEPPARAM_ACTIVATE_",
            "name": "Activate",
            "type": "Boolean",
            "value": "true"
          },
          {
            "id": 1455,
            "label": "_DB_STEPPARAM_PERSISTENT_",
            "name": "Persistent",
            "type": "Boolean",
            "value": "true"
          }
        ]
      }
    }
  ]
}
```

```

    }
  },
  {
    "id": 1273,
    "position": 2,
    "step": {
      "id": 1058,
      "params": [
        {
          "id": 1326,
          "label": "_DB_STEPPARAM_LOGEVENTS_",
          "name": "LogEvents",
          "type": "Boolean",
          "value": "true"
        }
      ]
    }
  }
]
}
}
}

```

### **/oprule/rule/{ruleId}/step/assignment/{assignmentId}**

This operation returns the available details of a specific step that is assigned to the operational rule.

### **Syntax**

GET /oprule/rule/{ruleId}/step/assignment/{assignmentId}

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule step assignment. This value is returned by the <code>GET /oprule/rule/{ruleId}/step/assignments</code> operation.

## Returns

Returns the class [StepAssignment](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.



Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

## Example

This example returns the details of a step assignment (ID 1272), that is the list of step parameters and their values as they are defined for the operational rule (ID 1066).

### Request URL

<https://localhost:1611/api/1/oprule/rule/1066/step/assignment/1272>

### Response Body

```
{
  "assignment": {
    "id": 1272,
    "position": 1,
    "step": {
      "id": 1087,
      "params": [
        {
          "id": 1453,
          "label": "_DB_STEPPARAM_MODULENAMES_",
          "name": "ModuleName",
          "type": "Enum",
          "value": "DeviceManagement"
        },
        {
          "id": 1454,
          "label": "_DB_STEPPARAM_ACTIVATE_",
          "name": "Activate",
          "type": "Boolean",
          "value": "true"
        },
        {
          "id": 1455,
          "label": "_DB_STEPPARAM_PERSISTENT_",
          "name": "Persistent",
          "type": "Boolean",
          "value": "true"
        }
      ]
    }
  }
}
```

**/oprule/icons**

This operation returns the list of icons that are available for the MyApps browser interface.

**Syntax**

GET /oprule/icons

**Parameters**

The action takes no parameters.

**Returns**

Returns the class *MyAppsIcons*.

**Comments**

**Possible Error Codes**

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

**Example**

This example returns the names of all icon files that can be used with the BCM application kiosk MyApps.

**Request URL**

https://localhost:1611/api/1/oprule/icons

**Response Body**

```

{
  "icons": [
    "or/administration.png",
    "or/alert.png",
    "or/analyse.png",
    "or/arrow_down.png",
    "or/arrow_up.png",
    "or/atom.png",
    "or/brush.png",
    "or/calendar.png",
    "or/calendar2.png",
    "or/cd.png",
    "or/cdburn.png",
    "or/chart.png",
    "or/chartline.png",
    "or/cleanup.png",
    ....
    "or/viewsmallicons.png",
    "or/viewtitles.png",
    "or/webcam.png"
  ]
}

```

### /oprule/step/classes

This operation returns the list of step classes that available for the operational rule.

### Syntax

GET /oprule/step/classes

### Parameters

The action takes no parameters.

### Returns

Returns the class [StepClasses](#).

### Comments

### Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.

Error Code	Description
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the list of all step classes currently available in BCM.

### Request URL

`https://localhost:1611/api/1/oprule/step/classes`

### Response Body

```
{
  "classes": [
    "_DB_STEPCLASS_AGENTCONFIG_",
    "_DB_STEPCLASS_CUSTOMINVENTORY_",
    "_DB_STEPCLASS_EVENTLOGMANAGER_",
    "_DB_STEPCLASS_FILEDIRHANDLING_",
    "_DB_STEPCLASS_HWINVENTORY_",
    "_DB_STEPCLASS_INVENTORY_",
    "_DB_STEPCLASS_MASTERSTEPS_",
    "_DB_STEPCLASS_MONITORING_",
    "_DB_STEPCLASS_PACKAGER_",
    "_DB_STEPCLASS_PATCHMANAGEMENT_",
    "_DB_STEPCLASS_POWERMANAGEMENT_",
    "_DB_STEPCLASS_PROCESMGT_",
    "_DB_STEPCLASS_REMOTEINVENTORY_",
    "_DB_STEPCLASS_SECURITYINVENTORY_",
    "_DB_STEPCLASS_SECURITYPRODUCTSMANAGEMENT_",
    "_DB_STEPCLASS_SWDISTRIBUTION_",
    "_DB_STEPCLASS_TOOLS_",
    "_DB_STEPCLASS_USERDIALOG_",
    "_DB_STEPCLASS_VIRTUALINFRASTRUCTUREMANAGEMENT_",
    "_DB_STEPCLASS_WINDOWS_",
    "_DB_STEPCLASS_WINDOWSDEVICEMANAGEMENT_",
    "_DB_STEPCLASS_XPFIREWALL_"
  ]
}
```

## /oprule/steps

This operation returns the list of steps that are available for operational rules. Filters can be applied to this search.

## Syntax

GET /oprule/steps?class=StepClassName&query=ExpressionToMatch&language=DisplayLanguage

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
class	query	no	integer	The unique BCM database identifier of the step class. This value is returned by the <code>/oprule/step/classes</code> operation.
query	query	no	integer	Enter into this field a specific expression or combination of words that are part of the name or contained the notes of the step you are looking for.
language	query	no	integer	The language in which the search will be carried out; this can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.

## Returns

Returns the class [StepList](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example lists all steps that are part of the Process Management (`_DB_STEPCLASS_PROCESSMGT_`) class and which contain the Brazilian expression "processo" in their localized user friendly name.

**Request URL**

```
https://localhost:1611/api/1/oprule/steps?
class=_DB_STEPCLASS_PROCESSMGT_&query=processo&language=Brasileiro
```

**Response Body**

```
{
  "steps": [
    {
      "id": 1128,
      "name": "_DB_STEPNAME_PROCESSBLACKLIST_",
      "class": "_DB_STEPCLASS_PROCESSMGT_",
      "notes": "_DB_STEPNOTE_PROCESSBLACKLIST_"
    },
    {
      "id": 1129,
      "name": "_DB_STEPNAME_PROCESSWHITELIST_",
      "class": "_DB_STEPCLASS_PROCESSMGT_",
      "notes": "_DB_STEPNOTE_PROCESSWHITELIST_"
    },
    {
      "id": 1086,
      "name": "_DB_STEPNAME_KILLPROCESSES_",
      "class": "_DB_STEPCLASS_PROCESSMGT_",
      "notes": "_DB_STEPNOTE_KILLPROCESSES_"
    },
    {
      "id": 1250,
      "name": "_DB_STEPNAME_WAITPROCESS_",
      "class": "_DB_STEPCLASS_PROCESSMGT_",
      "notes": "_DB_STEPNOTE_WAITPROCESS_"
    },
    {
      "id": 1281,
      "name": "_DB_STEPNAME_CHECKPROCESS_",
      "class": "_DB_STEPCLASS_PROCESSMGT_",
      "notes": "_DB_STEPNOTE_CHECKPROCESS_"
    }
  ]
}
```

**/oprule/step/{stepTypeId}**

This operation returns the list of parameters of a specific step.

**Syntax**

```
GET /oprule/step/{stepTypeId}
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
stepTypeld	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the step. This value is returned by the <code>/oprule/steps</code> operation.

## Returns

Returns the class [StepList](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example lists the parameters that are available for the step with the ID 1281 (\_DB\_STEPNAME\_CHECKPROCESS\_=Check Running Process) together with the necessary parameter information.

#### Request URL

<https://localhost:1611/api/1/oprule/step/1281>

#### Response Body

```
{
  "step": {
    "id": 1281,
    "params": [
      {
        "id": 2518,
        "label": "_DB_STEPPARAM_PROCESSNAME_",
        "name": "ProcessName",
        "type": "String"
      },
      {
        "id": 2519,
        "label": "_DB_STEPPARAM_STOPIFFOUND_",
        "name": "StopIfFound",
        "type": "Boolean"
      },
      {
        "id": 2520,
        "label": "_DB_STEPPARAM_SENDALEERT_",
        "name": "GenerateAlert",
        "type": "Boolean"
      },
      {
        "id": 2521,
        "label": "_DB_STEPPARAM_ALERTDESCRIPTION_CASE1_",
        "name": "AlertDescriptionNotFound",
        "type": "String"
      },
      {
        "id": 2522,
        "label": "_DB_STEPPARAM_ALERTSEVERITY_",
        "name": "AlertSeverity",
        "type": "Enum"
      },
      {
        "id": 2523,
```



```

    "label": "_DB_STEPPARAM_ALERTSUBCATEGORY_",
    "name": "AlertSubCategory",
    "type": "String"
  }
}
}

```

### /oprule/rule/{ruleId}/device/assignments

This operation returns the list of all devices that are assigned to a specific operational rule.

### Syntax

GET /oprule/rule/{ruleId}/device/assignments?brief=ShowDetails

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the GET /oprule/rules or a GET /oprule/folder/{folderId}/rules operation.

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns a summary of the assignments, that is their IDs, or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as errors, execution schedule, and so on.

## Returns

Returns the class *DeviceAssignmentAttrsArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example lists all assignments for an operational rule, that is all devices, device groups, users and/or user groups the rule is assigned to with the assignments information.

### Request URL

`https://localhost:1611/api/1/oprule/rule/1066/device/assignments?brief=false`

### Response Body

```
{
  "assignments": [
    {
```

```

    "id": 1011,
    "device": {
      "id": 1000,
      "name": "MySpyDevice",
      "groupName": ""
    },
    "schedule": {
      "id": 1248
    },
    "status": "_STATUS_EXECUTEDOK_",
    "transportMode": "Unicast",
    "enableTime": false,
    "isActive": true,
    "networkInstall": false,
    "executeWhileFails": false,
    "uploadStatusEveryExec": false,
    "runAsCurrentUser": false,
    "executionError": {
      "errorStepNumber": 0,
      "errorType": "_DB_OPRULEERRORTYPE_NONE_",
      "errorCode": 0,
      "errorText": "_ERRORCODE_0_"
    }
  }
]
}

```

### **/oprule/rule/device/assignment/{assignmentId}**

This operation returns the details of an assignment between a specific device and operational rule.

### **Syntax**

GET /oprule/rule/device/assignment/{assignmentId}

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and device. This value is returned by the <code>/oprule/rule/{ruleId}/device/assignments</code> operation.

## Returns

Returns the class *DeviceAssignmentAttrsObject*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

### Example

This example provides the details on a specific device assignment (ID 1011) of an operational rule.

#### Request URL

`https://localhost:1611/api/1/oprule/rule/device/assignment/1011`

#### Response Body

```
{
  "assignment": {
    "id": 1011,
    "device": {
      "id": 0,
      "group": {
        "id": 0
      }
    },
  },
  "schedule": {
    "id": 1248
  },
  "status": "_STATUS_EXECUTEDOK_",
  "transportMode": "Unicast",
  "specialCase": false,
  "bypassTransferWindows": false,
  "wakeUpDevices": false,
  "uploadStatus": true,
  "executeWhileFails": false,
  "uploadStatusEveryExec": false,
  "runAsCurrentUser": false,
  "executionError": {
    "errorStepNumber": 0,
    "errorType": "",
    "errorCode": 0,
    "errorText": ""
  }
}
```

**/oprule/rule/{ruleId}/device/group/assignments**

This operation returns the list of all device groups that are assigned to a specific operational rule.

**Syntax**

GET /oprule/rule/{ruleId}/device/group/assignments?brief=ShowDetails

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the GET /oprule/rules or a GET /oprule/folder/{folderId}/rules operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns a summary of the assignments, that is their IDs, or also other available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as errors, execution schedule, and so on.

## Returns

Returns the class *DeviceGroupAssignmentAttrsArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the list of device groups to which the rule is assigned to with their respective information.

### Request URL

```
https://localhost:1611/api/1/oprule/rule/1066/device/group/assignments?brief=false
```

### Response Body

```
{
  "assignments": [
    {
      "id": 1003,
      "deviceGroup": {
        "id": 255,
        "name": "All my devices"
      },
      "schedule": {
        "id": 1250
      },
      "status": "_STATUS_ASSIGNPAUSED_",
      "enableTime": false,
      "isActive": true,
      "networkInstall": false,
    }
  ]
}
```

```
    "executeWhileFails": false,
    "uploadStatusEveryExec": false,
    "runAsCurrentUser": false
  }
]
```

**/oprule/rule/device/group/assignment/{assignmentId}**

This operation returns the details of an assignment between a specific device group and operational rule.

**Syntax**

GET /oprule/rule/device/group/assignment/{assignmentId}

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a device group. This value is returned by the GET /oprule/rule/{ruleId}/device/group/assignments operation.



## Returns

Returns the class *DeviceGroupAssignmentAttrsObject*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

The following subfolders are listed with further information and their hierarchy after only the out-of-the-box objects are imported and no other folders or rules are created yet.

### Request URL

```
https://localhost:1611/api/1/oprule/rule/device/group/assignment/1003
```

### Response Body

```
{
  "assignment": {
    "id": 1003,
    "deviceGroup": {
      "id": 0
    },
  },
  "schedule": {
    "id": 1250
  },
  "status": "_STATUS_ASSIGNPAUSED_",
  "specialCase": false,
  "bypassTransferWindows": false,
  "wakeupDevices": false,
  "uploadStatus": true,
}
```

```
"executeWhileFails": false,
"uploadStatusEveryExec": false,
"runAsCurrentUser": false
}
```

**/oprule/rule/{ruleId}/user/assignments**

This operation returns the list of all users that are assigned to a specific operational rule.

**Syntax**

GET /oprule/rule/{ruleId}/user/assignments?brief=ShowDetails

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the GET /oprule/rules or a GET /oprule/folder/{folderId}/rules operation.

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns a summary of the assignments, that is their IDs, or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as errors, execution schedule, and so on.

## Returns

Returns the class *UserAssignmentAttrsArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example lists all users to which the rule (ID 1066) is assigned to with their respective information.

### Request URL

```
https://localhost:1611/api/1/oprule/rule/1066/user/assignments?brief=false
```

### Response Body

```
{
  "assignments": [
    {
      "id": 1000,
      "user": {
```

```
    "id": 1002,
    "name": "SysAdmin",
    "group": {
      "id": 0,
      "name": ""
    }
  },
  "schedule": {
    "id": 1252
  },
  "status": "_STATUS_ASSIGNWAITING_",
  "createTime": -921101610,
  "networkInstall": false,
  "installAsUser": true,
  "userAssignPolicy": "_ASSIGN_USERLOGIN_"
}
]
```

### **/oprule/rule/user/assignment/{assignmentId}**

This operation returns the details of an assignment between a specific user and operational rule.

### **Syntax**

GET /oprule/rule/user/assignment/{assignmentId}

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and user. This value is returned by the <code>/oprule/rule/{ruleId}/user/assignments</code> operation.

## Returns

Returns the class *UserAssignmentAttrsObject*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

### Example

The following subfolders are listed with further information and their hierarchy after only the out-of-the-box objects are imported and no other folders or rules are created yet.

#### Request URL

```
https://localhost:1611/api/1/oprule/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&count=100
```

#### Response Body

### /oprule/rule/{ruleId}/user/group/assignments

This operation returns the list of all user groups that are assigned to a specific operational rule.

### Syntax

```
GET /oprule/rule/{ruleId}/user/group/assignments?brief=ShowDetails
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns a summary of the assignments, that is their IDs, or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as errors, execution schedule, and so on.

## Returns

Returns the class *UserGroupAssignmentAttrsArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example returns the list of user groups to which the operational rule (ID 1066) is assigned with all respective information.

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1066/user/group/assignments?brief=false
```

#### Response Body

```
{
  "assignments": [
    {
      "id": 1000,
      "userGroup": {
        "id": 246,
        "name": "Users with advertised rules"
      },
      "schedule": {
        "id": 1251
      },
      "status": "_STATUS_ASSIGNWAITING_",
      "createTime": -921101596,
      "networkInstall": false,
      "installAsUser": true,
      "userAssignPolicy": "_ASSIGN_USERLOGIN_"
    }
  ]
}
```

### /oprule/rule/user/group/assignment/{assignmentId}

This operation returns the details of an assignment between a specific user group and operational rule.

### Syntax

```
GET /oprule/rule/user/group/assignment/{assignmentId}
```



## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a specific user group. This value is returned by the <code>GET /oprule/rule/{ruleId}/user/group/assignments</code> operation.

## Returns

Returns the class [UserGroupAssignmentAttrsArray](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

### Example

The following subfolders are listed with further information and their hierarchy after only the out-of-the-box objects are imported and no other folders or rules are created yet.

#### Request URL

```
https://localhost:1611/api/1/oprule/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&count=100
```

#### Response Body

### /oprule/rule/{ruleId}/dependencies

This operation returns the list of operational rules that are dependent on another specific rule.

### Syntax

```
GET /oprule/rule/{ruleId}/dependencies
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

## Returns

Returns the class *OpRuleDependencies*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example returns the list of operational rules on which the execution of the rule is dependent on, together with the respective information for each.

#### Request URL

```
https://localhost:1611/api/1/oprule/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&count=100
```

#### Response Body

### /oprule/rule/{ruleId}/packages

This operation returns the list of packages assigned to a specific operational rule.

### Syntax

```
GET /oprule/rule/{ruleId}/packages?brief=ShowDetails
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the packages or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.

## Returns

Returns the class *PackageAssignment*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example lists all packages that are added to the operational rule (ID 1027) together with its detail information.

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1027/packages?brief=false
```

#### Response Body

```
{
  "packages": [
    {
      "id": 1000,
      "name": "NmapFiles.cst",
      "type": "CUSTOM"
    }
  ]
}
```

### /schedule/{scheduleId}

This operation returns the details of the operational rule assignment schedule.

### Syntax

```
GET /schedule/{scheduleId}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scheduleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule's schedule. This value is returned, depending on the assigned object - by either the <i>/oprule/rule/device/assignment/{assignmentId}</i> , <i>/oprule/rule/device/group/assignment/{assignmentId}</i> , <i>/oprule/rule/user/assignment/{assignmentId}</i> , or the <i>/oprule/rule/user/group/assignment/{assignmentId}</i> operation.

## Returns

Returns the class [ScheduleResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

### Example

This example returns the information of a specific schedule (ID 1248) for the execution of operational rule 1066 on the master device (ID 1000, assignment ID 1011).

#### Request URL

`https://localhost:1611/api/1/schedule/1248`

#### Response Body

```
{
  "schedule": {
    "id": 1248,
    "name": "Activate Peripheral logging",
    "validity": {
      "execution": "immediate",
      "executionAt": 0,
      "termination": "runCount",
      "terminationRunCount": 1,
      "terminationAt": 0
    },
    "frequency": {
      "type": "schedule",
      "scheduleMonths": [
        "*"
      ],
      "period": "1min",
      "periodAt": "",
      "periodFrom": "*:00",
      "periodTo": "1:00",
      "scheduleType": "daily",
      "cron": 0
    }
  }
}
```



## PUT Operations

### /oprule/folder/{folderId}/subfolder

This operation creates a new operational rule subfolder in a specific operational rules folder.

### Syntax

```
PUT /oprule/folder/{folderId}/subfolder
```

The body must contain the definitions of the `FolderAttrs` class.

```
{ "name": "string" "notes": "string" }
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rules subfolder. This value is returned by either the <code>/oprule/subfolders</code> or another <code>/oprule/folder/{folderId}/subfolders</code> operation.

Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	FolderAttrs	This array contains the data for the new folder to be created.

## Returns

Returns the class *OperationalRuleFolderArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16040	Bad object type.
16041	Bad object name.
16043	The supplied attribute name does not exist.

### Example

The following example creates a new folder directly under the folder "My new folder" (ID 256) and returns its ID:

#### Request URL

```
https://localhost:1611/api/1/oprule/folder/256/subfolder
```

The body must contain the folder information, for example:

```
{ "name": "My 2nd new folder", "notes": "Another folder for testing purposes" } }
```

#### Response Body

```
{
  "folder": {
    "id": 257
  }
}
```

### /oprule/rule

This operation creates a new operational rules directly under the Operational Rules top node.

### Syntax

```
PUT /oprule/rule
```

The body must contain the definitions of the `OperationalRuleAttrs` class.

```
{ "name": "string" "type": "string" "url": "string" "notes": "string" "kioskIcon":  
"string" }
```

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
OperationalRuleAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	OperationalRuleAttrs	This array contains the data for the new operational rule to be created.

Returns

Returns the class `PutRuleResult`.

Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16040	Bad object type.
16041	Bad object name.
16043	The supplied attribute name does not exist.

## Example

The following example creates a new operational rule of type "Quick Link" under the main **Operational Rules** node and returns the Id of the new rule:

### Request URL

```
https://localhost:1611/api/1/oprule/folder/256/subfolder
```

The body must contain the folder information, for example:

```
{ "name": "Quick Link Test", "type": "Quick Link", "url": "www.google.com", "notes":  
"string", "kioskIcon": "or/Kiosk/globe.png" } }
```

**Response Body**

```
{
  "rule": {
    "id": 1067
  }
}
```

**/oprule/folder/{folderId}/rule**

This operation creates a new operational rule in a specific folder.

**Syntax**

PUT /oprule/folder/{folderId}/rule

The body must contain the definitions of the `OperationalRuleAttrs` class.

```
{ "name": "string" "type": "string" "url": "string" "notes": "string" "kioskIcon":
  "string" }
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rules subfolder. This value is returned by either the <i>/oprule/subfolders</i> or another <i>/oprule/folder/{folderId}/subfolders</i> operation.

Name	Type	Mand.	Data Type	Description
OperationalRuleAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	OperationalRuleAttrs	This array contains the data for the new operational rule to be created.

## Returns

Returns the class [PutRuleResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.



Error Code	Description
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16040	Bad object type.
16041	Bad object name.
16043	The supplied attribute name does not exist.

## Example

The following example creates a new standard operational rule under the folder "My 2nd new folder" (ID 257) and returns its Id:

### Request URL

```
https://localhost:1611/api/1/oprule/folder/257/rule
```

The body must contain the basic information of the new rule, for example:

```
{ "name": "My new Test Rule", "type": "Operational Rule", "url": "string", "notes": "A rule to make some tests", "kioskIcon": "string" } }
```

### Response Body

```
{
  "rule": {
    "id": 1068
  }
}
```

**/oprule/folder/{folderId}**

This operation updates the data of a specific operational rules folder.

**Syntax**

PUT /oprule/folder/{folderId}

The body must contain the definitions of the FolderAttrs class.

```
{ "name": "string" "notes": "string" }
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rules folder. This value is returned by either the /oprule/subfolders or another /oprule/folder/{folderId}/subfolders operation.

Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	FolderAttrs	This array contains the new data of folder to be updated.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16043	The supplied attribute name does not exist.

## Example

The following subfolders are listed with further information and their hierarchy after only the out-of-the-box objects are imported and no other folders or rules are created yet.

### Request URL

```
https://localhost:1611/api/1/oprule/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&count=100
```

### Response Body

## /oprule/rule/{ruleId}

This operation updates the data of a specific operational rule.

## Syntax

```
PUT /oprule/rule/{ruleId}
```

The body must contain the definitions of the `OperationalRuleAttrsForUpdate` class.

```
{ "name": "string" "type": "string" "url": "string" "notes": "string" "kioskIcon":
"string" }
```

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
OperationalRuleAttrsBody	Body	Optional	OperationalRuleAttrs	This array contains the new data for the operational rule to be updated.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16043	The supplied attribute name does not exist.

### Example

The following subfolders are listed with further information and their hierarchy after only the out-of-the-box objects are imported and no other folders or rules are created yet.

#### Request URL

```
https://localhost:1611/api/1/oprule/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&count=100
```

#### Response Body

### /oprule/rule/{ruleId}/step/{stepTypeId}

This operation assigns a step to a specific operational rule.

### Syntax

```
PUT /oprule/rule/{ruleId}/step/{stepTypeId}
```

The body must contain the definitions of the `StepAttrs` class.

```
"stopCondition": "string", "verificationCondition": "string", "notes": "string",
"params": [ { "id": "int", "value": "string" } ] }
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.



Name	Type	Mand.	Data Type	Description
stepTypeld	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the step. This value is returned by the <code>/oprule/steps</code> operation.

Name	Type	Mand.	Data Type	Description
stepAttributes	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	StepAttrs	This array contains the necessary data to assign the specific step to the operational rule.

## Returns

Returns the class *PutRuleStepAssignmentResult*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16043	The supplied attribute name does not exist.

## Example

The following example assigns and instance of step **Check Running Process** (ID 1281) to rule "My new Test Rule" (ID 1068) and returns the ID of the rule step assignment.

### Request URL

```
https://localhost:1611/api/1/oprule/folder/257/rule
```

The body must contain the basic information of the step, for example:

```
{ "stopCondition": "_DB_STEPACTION_CONTINUE_", "verificationCondition":
"_DB_STEPACTION_VERIFYLOOPONERROR_", "notes": "My test step", "params": [ { "id":
"2518", "value": "Explorer.EXE" } { "id": "2519", "value": "true" } { "id": "2520",
"value": "true" } { "id": "2521", "value": "My Alert" } { "id": "2522", "value":
"Warning" } { "id": "2523", "value": "Process" } ] }
```

### Response Body

```
{
  "assignment": {
    "id": 1354
  }
}
```

## /oprule/rule/{ruleId}/device/{deviceId}

This operation assigns a device to a specific operational rule.

## Syntax

```
PUT /oprule/rule/{ruleId}/device/{deviceId}?activation=ActivationMode
```

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
activation	query	no	List[string]	Select from this drop-down box if the assignment is to be effected automatically or manually.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

## Example

The following example assigns the rule "My test rule" (ID 1068) to the master (ID 1000) and returns the assignment Id:

### Request URL

```
https://localhost:1611/api/1/oprule/rule/1068/device/1000?activation=automatic
```

### Response Body

```
{
  "assignment": {
    "id": 1015
  }
}
```

**/oprule/rule/{ruleId}/device/group/{groupId}**

This operation assigns a device group to a specific operational rule.

**Syntax**

PUT /oprule/rule/{ruleId}/device/group/{groupId}?activation=ActivationMode

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the GET /oprule/rules or a GET /oprule/folder/{folderId}/rules operation.

Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the device group to assign. This value is returned by the <code>GET /group/subgroups</code> , <code>GET /group/{groupId}/hierarchy</code> and <code>GET /group/subgroup/{groupId}/subgroups</code> operations.
activation	query	no	List[string]	Select from this drop-down box if the assignment is to be effected automatically or manually.

## Returns

Returns the class [AssignmentDetails](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.



Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

### Example

The following example assigns the rule "My test rule" (ID 1068) to group "All my devices" (ID 255) with manual activation and returns the assignment Id:

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1068/device/group/255?activation=manual
```

#### Response Body

```
{
  "assignment": {
    "id": 1004
  }
}
```

**/oprule/rule/{ruleId}/user/group/{groupId}**

This operation assigns a user group to a specific operational rule.

**Syntax**

PUT /oprule/rule/{ruleId}/user/group/{groupId}

The body must contain the definitions of the `UserAssignmentAttrsPut` class.

```
{ "installAsUser": "boolean" "modeMyApps": "boolean" "notOnlyPrimaryUser": "boolean"
"userAssignPolicy": "string" }
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the user group. User group IDs are returned by the <code>GET /user/subgroups</code> or <code>GET /user/group/{groupId}/subgroups</code> operation.
UserAssignmentAttributes	query	no	[ <a href="#">UserAssignmentAttributes</a> ]	This array contains the data for the new operational rule assignment to be created.

## Returns

Returns the class [AssignmentDetails](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

### Example

The following example assigns the rule "My test rule" (ID 1068) to group "All users" (ID 178) and returns the assignment Id:

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1068/user/group/178
```

The body must contain the basic information for the assignment, for example:

```
{ { "installAsUser": "false", "modeMyApps": "false", "notOnlyPrimaryUser": "true",
"userAssignPolicy": "_ASSIGN_DEFAULTSCHEDULE_" } }
```

#### Response Body

```
{
  "assignment": {
    "id": 1002
  }
}
```

## /oprule/rule/{ruleId}/user/{userId}

This operation assigns a user to a specific operational rule.

### Syntax

PUT /oprule/rule/{ruleId}/user/{userId}

The body must contain the definitions of the `UserAssignmentAttrsPut` class.

```
{ "installAsUser": "boolean" "modeMyApps": "boolean" "notOnlyPrimaryUser": "boolean"
"userAssignPolicy": "string" }
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
userId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the user. User IDs are returned by the <code>GET /user/users</code> or <code>GET /user/group/{groupId}/users</code> operation.
UserAssignmentAttributes	Query	no	[ <a href="#">UserAssignmentAttributes</a> ]	This array contains the data for the new operational rule assignment to be created.

## Returns

Returns the class [AssignmentDetails](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

### Example

The following example assigns the rule "My test rule" (ID 1068) to user sysadmin (ID 1004) and returns the assignment Id:

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1068/user/group/178
```

The body must contain the basic information for the assignment, for example:

```
{ "installAsUser": "false", "modeMyApps": "false", "notOnlyPrimaryUser": "true",
  "userAssignPolicy": "_ASSIGN_DEFAULTSCHEDULE_" }
```

#### Response Body

```
{
  "assignment": {
    "id": 1002
  }
}
```

**/oprule/rule/{ruleId}/step/assignment/{assignmentId}**

This operation updates the parameters of a step assignment of a specific operational rule.

**Syntax**

PUT /oprule/rule/{ruleId}/step/assignment/{assignmentId}

The body must contain the definitions of the `StepAssignmentForUpdate` class.

```
{ "position": "int" "params": "array" }
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.



Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule step assignment. This value is returned by the <code>GET /oprule/rule/{ruleId}/step/assignments</code> operation.

Name	Type	Mand.	Data Type	Description
StepAssignmentForUpdate	Update	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	[ <a href="#">StepAssignmentForUpdate</a> ]	This array contains the new data for the operational rule step update.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16055	The operational rule can not be modified while it is waiting to be sent.

### Example

This example modifies the check agent version step assignment to test for version 12..

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1069/step/assignment/1378
```

The body must contain the parameters to be updated, for example:

```
{ "params": [ { "id": "1099", "value": "12.0" } ] }
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /oprule/rule/{ruleId}/device/{deviceId}

This operation assigns a device to a specific operational rule.

### Syntax

```
PUT /oprule/rule/{ruleId}/device/{deviceId}?activation=ActivationMode
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
deviceid	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database ID of the device. This value is returned by either the <code>GET /devices</code> or the <code>GET /device</code> operation.
activation	query	no	List[string]	Select from this drop-down box if the assignment is to be effected automatically or manually.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

## Example

The following example assigns the rule "My test rule" (ID 1068) to the master (ID 1000) and returns the assignment Id:

### Request URL

```
https://localhost:1611/api/1/oprule/rule/1068/device/1000?activation=automatic
```

### Response Body

```
{
  "assignment": {
    "id": 1015
  }
}
```

## /oprule/rule/{ruleId}/device/group/{groupId}

This operation assigns a device group to a specific operational rule.

### Syntax

PUT /oprule/rule/{ruleId}/device/group/{groupId}?activation=ActivationMode

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the GET /oprule/rules or a GET /oprule/folder/{folderId}/rules operation.

Name	Type	Mand.	Data Type	Description
groupId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the device group to assign. This value is returned by the <code>GET /group/subgroups</code> , <code>GET /group/{groupId}/hierarchy</code> and <code>GET /group/subgroup/{groupId}/subgroups</code> operations.
activation	query	no	List[string]	Select from this drop-down box if the assignment is to be effected automatically or manually.

## Returns

Returns the class [AssignmentDetails](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.



Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

### Example

The following example assigns the rule "My test rule" (ID 1068) to group "All my devices" (ID 255) with manual activation and returns the assignment Id:

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1068/device/group/255?activation=manual
```

#### Response Body

```
{
  "assignment": {
    "id": 1004
  }
}
```

**/oprule/rule/user/assignment/{assignmentId}**

This operation updates the parameters of the assignment between an operational rule and a user.

**Syntax**

PUT /oprule/rule/user/assignment/{assignmentId}

The body must contain the definitions of the `UserAssignment` class.

```
{ "status": "string" "scheduleId": "int" "installAsUser": "boolean"
"notOnlyPrimaryUser": "boolean" "userAssignPolicy": "boolean" "networkInstall":
"boolean" "bypassTransferWindows": "boolean" "wakeupDevices": "boolean"
"executeWhileFails": "boolean" "uploadStatus": "boolean" "uploadStatusEveryExec":
"boolean" }
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a specific user. This value is returned by the <code>GET /oprule/rule/{ruleId}/user/assignments</code> operation.

Name	Type	Mand.	Data Type	Description
UserAssignment	body	no	[ <a href="#">UserAssignment</a> ]	This class contains the new data for the operational rule assignment update.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16043	The supplied attribute name does not exist.

## Example

The following example updates the user assignment to install the rule as the user and to use a transfer window.

### Request URL

```
https://localhost:1611/api/1/oprule/rule/user/assignment/1009
```

The body must contain the parameters to be updated, for example:

```
{ "installAsUser": "true", "bypassTransferWindows": "false" }
```

### Response Body

```
{
  "errorCode": 0
}
```

### /oprule/rule/user/group/assignment/{assignmentId}

This operation updates the parameters of the assignment between an operational rule and a user group.

### Syntax

PUT /oprule/rule/user/group/assignment/{assignmentId}

The body must contain the definitions of the `UserGroupAssignment` class.

```
{ "status": "string" "scheduleId": "int" "installAsUser": "boolean"
  "notOnlyPrimaryUser": "boolean" "userAssignPolicy": "boolean" "networkInstall":
  "boolean" "bypassTransferWindows": "boolean" "wakeupDevices": "boolean"
  "executeWhileFails": "boolean" "uploadStatus": "boolean" "uploadStatusEveryExec":
  "boolean" }
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a specific user group. This value is returned by the <code>GET /oprule/rule/{ruleId}/user/group/assignments</code> operation.
UserGroupAssignment	body	no	<a href="#">[UserGroupAssignment]</a>	This class contains the new data for the operational rule assignment update.

## Returns

Returns the class [UserGroupAssignmentAttrsArray](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16043	The supplied attribute name does not exist.

### Example

The following example updates the user assignment to install the rule as the user and to use a transfer window.

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/user/group/assignment/1000
```

The body must contain the parameters to be updated, for example:

```
{ "installAsUser": "true", "bypassTransferWindows": "false" }
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /oprule/rule/{ruleId}/dependency/{dependentRuleId}

This operation creates a new or updates an existing dependency between operational rules. It is used to create a rule dependency on one rule on another by providing rule id's in specified format. Dependency order generated will be last one in the list of dependencies. This operation can also be used to update the dependency order by providing new position number, provided rule was created earlier without providing position.

### Syntax

```
PUT /oprule/rule/{ruleId}/dependency/{dependentRuleId}
```

The body must contain the definitions of the `UserGroupAssignment` class.

```
{ "position": "int" }
```

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
dependentRuleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule which is dependend on the execution of the first rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.
OpRuleDependencyPosition	body	no	OpRuleDependencyPosition	This new dependency order position for this rule. This parameter is only applicable when updating. Do not provide a position number when creating a dependency.

## Returns

Returns the class *OpRuleDependencyOrder*.

## Comments

## Possible Error Codes

The following errors can be returned:



Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

## Example

The following example creates a dependency between a software distribution rule and the check agent version rule, that is, the distribution rule is dependent on the check version rule, because the package is only to be installed if the agent version is 12.

### Request URL

`https://localhost:1611/api/1/oprule/rule/1069/dependency/1032`

### Response Body

```
{
  "dependency": {
    "position": 1
  }
}
```

**/oprule/rule/{ruleId}/package/{packageId}**

This operation adds a package to a specific operational rule.

**Syntax**

PUT /oprule/rule/{ruleId}/package/{packageId}

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the GET /oprule/rules or a GET /oprule/folder/{folderId}/rules operation.
packageId	query	no	boolean	The unique BCM database identifier of the package. This value is returned by either the GET /packages/packages or a GET /packages/folder/{folderId}/packages operation.

## Returns

Returns the class *OperationResult*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16028	One or more objects could not be assigned.
16053	The operational rule does not have any steps.
16071	The number of objects permitted by the BMC Client Management license was exceeded.
16073	The number of devices to assign is higher than the remaining BMC Client Management licenses.
16074	The BMC Client Management license has expired.
16161	The operational rule contains steps which are no longer available with the current license.

## Example

The following example adds a package to the check agent version rule to be installed if the version is 12.

**Request URL**

```
https://localhost:1611/api/1/oprule/rule/1069/package/1001
```

**Response Body**

```
{  
  "errorCode": 0  
}
```

**/schedule/{scheduleId}**

This operation updates the parameters of the operational rule's schedule.

**Syntax**

```
PUT /schedule/{scheduleId}
```

The body must contain the definitions of the `Schedule` class.

```
{ "name": "string" "validity": "{ScheduleValidity}" "frequency":  
  "{ScheduleFrequency}" }
```

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scheduleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule's schedule. This value is returned, depending on the assigned object - by either the <i>/oprule/rule/device/assignment/{assignmentId}</i> , <i>/oprule/rule/device/group/assignment/{assignmentId}</i> , <i>/oprule/rule/user/assignment/{assignmentId}</i> , or the <i>/oprule/rule/user/group/assignment/{assignmentId}</i> operation.
Schedule	body	no	[ <a href="#">Schedule</a> ]	This class contains the new data for the operational rule schedule update.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16043	The supplied attribute name does not exist.

### Example

The following example updates the schedule of the check agent version rule from an immediate execution to execution at the next startup of the device.

#### Request URL

```
https://localhost:1611/api/1/schedule/1262
```

The body must contain the parameters to be updated, for example:

```
{ "validity": { "execution": "nextStartup" } }
```

#### Response Body

```
{
  "errorCode": 0
}
```

## DELETE Operations

### /oprule/folder/{folderId}

This operation deletes a specific operational rules folder.

### Syntax

```
DELETE /oprule/folder/{folderId}
```

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rules subfolder. This value is returned by either the <i>/oprule/subfolders</i> or another <i>/oprule/folder/{folderId}/subfolders</i> operation.

## Returns

Returns the class *OperationResult*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16042	The object can not be deleted as it is currently in use.
16046	The device is linked to objects.

### Example

The following example deletes the operational rule test folder (ID 258).

#### Request URL

```
https://localhost:1611/api/1/oprule/folder/258
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /oprule/rule/{ruleId}/step/assignment/{assignmentId}

This operation deletes a specific operational rule step assignment.

### Syntax

```
DELETE /oprule/rule/{ruleId}/step/assignment/{assignmentId}
```

### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule step assignment. This value is returned by the <code>GET /oprule/rule/{ruleId}/step/assignments</code> operation.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

The following example removes step check agent version (ID 1015) from the check agent version rule (ID 1069).

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1069/step/assignment/1378
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /oprule/rule/{ruleId}

This operation deletes a specific operational rule.

### Syntax

```
DELETE /oprule/rule/{ruleId}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16042	The object can not be deleted as it is currently in use.

### Example

The following example deletes the check agent version rule (ID 1069).

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1069
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /oprule/rule/device/assignment/{assignmentId}

This operation deletes a specific assignment between a device and an operational rule.

### Syntax

```
DELETE /oprule/rule/device/assignment/{assignmentId}?activation=ActivationMode
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a device. This value is returned by the <code>/oprule/rule/{ruleId}/device/assignments</code> operation.
activation	query	no	List[string]	Select from this drop-down box if the unassignment is to be effected automatically or manually.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16056	The rule assignment has been deleted.
16057	The rule has been automatically unassigned.
16116	The object could not be found.
16163	The operational rule was assigned to the device via a user. To unassign it you need to unassign the user from the operational rule.
16164	The device assignment could not be removed because it was effected via a device group. You must remove the device group assignment.
16165	The user assignment could not be removed because it was effected via a user group. You must remove the user group assignment.

## Example

The following example deletes the assignment (ID 1018) between the check agent version rule (ID 1069) and its assigned device (ID 1000) with automatic unassignment activation.

### Request URL

```
https://localhost:1611/api/1/oprule/rule/device/assignment/1018?activation=automatic
```

### Response Body

```
{
  "errorCode": 0
}
```

**/oprule/rule/device/group/assignment/{assignmentId}**

This operation deletes a specific assignment between a device group and an operational rule.

**Syntax**

DELETE /oprule/rule/device/group/assignment/{assignmentId}?activation=ActivationMode

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a device group. This value is returned by the GET /oprule/rule/{ruleId}/device/group/assignments operation.
activation	query	no	List[string]	Select from this drop-down box if the unassignment is to be effected automatically or manually.

**Returns**

Returns the class *OperationResult*.



## Comments

### Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
11	An object is already registered under this name.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
14	Incorrect parameter value. Please verify the values of the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16043	The supplied attribute name does not exist.

### Example

The following example deletes the assignment (ID 1018) between the check agent version rule (ID 1069) and its assigned device group (ID 255) with automatic unassignment activation.

#### Request URL

<https://localhost:1611/api/1/oprule/rule/device/group/assignment/1005?activation=automatic>

#### Response Body

```
{
  "errorCode": 0
}
```

**/oprule/rule/user/assignment/{assignmentId}**

This operation deletes a specific assignment between a user and an operational rule.

**Syntax**

DELETE /oprule/rule/user/assignment/{assignmentId}

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a specific user. This value is returned by the GET /oprule/rule/{ruleId}/user/assignments operation.

**Returns**

Returns the class *OperationResult*.

**Comments**

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16056	The rule assignment has been deleted.
16057	The rule has been automatically unassigned.
16116	The object could not be found.
16163	The operational rule was assigned to the device via a user. To unassign it you need to unassign the user from the operational rule.
16164	The device assignment could not be removed because it was effected via a device group. You must remove the device group assignment.
16165	The user assignment could not be removed because it was effected via a user group. You must remove the user group assignment.

## Example

The following example deletes the assignment (ID 1009) between the check agent version rule (ID 1069) and its assigned user (ID 1002).

### Request URL

```
https://localhost:1611/api/1/oprule/rule/user/assignment/1009
```

### Response Body

```
{
```

```
"errorCode": 0
}
```

**/oprule/rule/user/group/assignment/{assignmentId}**

This operation deletes a specific assignment between a user group and an operational rule.

**Syntax**

DELETE /oprule/rule/user/group/assignment/{assignmentId}

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the specific operational rule and a specific user group. This value is returned by the GET /oprule/rule/{ruleId}/user/group/assignments operation.

**Returns**

Returns the class *OperationResult*.

## Comments

### Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16056	The rule assignment has been deleted.
16057	The rule has been automatically unassigned.
16116	The object could not be found.
16163	The operational rule was assigned to the device via a user. To unassign it you need to unassign the user from the operational rule.
16164	The device assignment could not be removed because it was effected via a device group. You must remove the device group assignment.
16165	The user assignment could not be removed because it was effected via a user group. You must remove the user group assignment.

### Example

The following example deletes the assignment (ID 1000) between the check agent version rule (ID 1069) and its assigned user group (ID 178).

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/user/group/assignment/1009
```

Response Body

```
{
  "errorCode": 0
}
```

**/oprule/rule/{ruleId}/dependency/{dependentRuleId}**

This operation deletes a specific operational rule dependency.

Syntax

```
DELETE /oprule/rule/{ruleId}/dependency/{dependentRuleId}
```

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

Name	Type	Mand.	Data Type	Description
dependentRuleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule which is dependend on the execution of the first rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16055	The operational rule can not be modified while it is waiting to be sent.

### Example

The following example deletes the dependency between the check agent version rule (ID 1069) and the software distribution rule (ID 1032).

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1032/dependency/1069
```

#### Response Body

```
{
  "errorCode": 0
}
```

### **/oprule/rule/{ruleid}/package/{packageid}**

This operation removes a package from a specific operational rule.

### Syntax

```
DELETE /oprule/rule/{ruleid}/package/{packageid}
```

### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
ruleId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the operational rule. This value is returned by either the <code>GET /oprule/rules</code> or a <code>GET /oprule/folder/{folderId}/rules</code> operation.
packageId	query	no	boolean	The unique BCM database identifier of the package. This value is returned by either the <code>GET /packages/packages</code> or a <code>GET /packages/folder/{folderId}/packages</code> operation.

## Returns

Returns the class [OperationResult](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16055	The operational rule can not be modified while it is waiting to be sent.

### Example

The following example removes the package (ID 1001) from the check agent version rule (ID 1069).

#### Request URL

```
https://localhost:1611/api/1/oprule/rule/1069/package/1001
```

#### Response Body

```
{
  "errorCode": 0
}
```

## Operational Rule Return Classes

The following classes can be returned by the different operational rule actions:

Class	Description
<a href="#"><i>AssignmentDetails</i></a>	Provides operational rule - target assignment information.
<a href="#"><i>AssignmentId</i></a>	Provide ID of operational rule - target assignment.
<a href="#"><i>AssignmentValuesResult</i></a>	Provides step parameter values for step assignment.
<a href="#"><i>Device</i></a>	Returns the search result for one specific device.
<a href="#"><i>DeviceAssignment</i></a>	Provides operational rule device assignment update information.

Class	Description
<i>DeviceAssignmentAttrs</i>	Provides information about a specific operational rule device assignment.
<i>DeviceAssignmentAttrsArray</i>	Provides the list of device assignments.
<i>DeviceAssignmentAttrsObject</i>	Provides the list of operational rule device assignments.
<i>DeviceAssignmentObj</i>	Provides specific operational rule device assignment information.
<i>DeviceGroup</i>	Provides device group specific information
<i>DeviceGroupAssignment</i>	Provides operational rule device group assignment update information.
<i>DeviceGroupAssignmentAttrs</i>	Provides information about a specific operational rule device group assignment.
<i>DeviceGroupAssignmentAttrsArray</i>	Provides the list of device group assignments.
<i>DeviceGroupAssignmentAttrsObject</i>	Provides the list of operational rule device group assignments.
<i>DeviceGroupAssignmentObj</i>	Provides specific operational rule device group assignment information.
<i>ExecutionError</i>	Provides operational rule execution errors.
<i>FolderAttrs</i>	Provides information required to create or update an operational rules folder.
<i>FolderValuesResult</i>	Provides operational rules subfolder ID.
<i>MyAppsIcons</i>	Provides list of icons available for MyApps.
<i>OperationalRuleArray</i>	Returns rule count and list of rule values.
<i>OperationalRuleAttrs</i>	Provides values for operational rule creation.
<i>OperationalRuleAttrsForUpdate</i>	Provides values for operational rule update.
<i>OperationalRuleFolderArray</i>	Returns operational rule folder count and a list of children's values.
<i>OperationalRuleFolderValues</i>	Provides the list of folder children and their values.
<i>OperationalRuleFolderValuesArray</i>	Provides the information available for an operational rules folder.
<i>OperationalRuleValues</i>	Return operational rules folder information.
<i>OperationalRuleValuesArray</i>	Returns the information that is available for an individual operational rule.
<i>OperationalRuleValuesResult</i>	Returns the .

Class	Description
<i>OperationResult</i>	Returns the result (error code) of the operation.
<i>OpRuleDependencies</i>	Returns the list of dependent operational rules on a specific rule.
<i>OpRuleDependencyDetails</i>	Returns information about the dependent operational rules of a specific rule.
<i>OpRuleDependencyOrder</i>	Returns order of the dependent operational rules of a specific rule.
<i>OpRuleDependencyPosition</i>	Returns the position number of a specific dependent rule.
<i>OpRuleDependencyResult</i>	Returns information about the dependent operational rules of a specific rule.
<i>PackageAssignment</i>	Returns the list of packages assigned to a specific operational rule.
<i>PackageAssignmentDetails</i>	Returns information a specific package.
<i>PutRuleFolderResult</i>	Returns the result of an operational rule folder creation operation.
<i>PutRuleResult</i>	Returns the result of an operational rule creation operation.
<i>PutRuleStepAssignmentResult</i>	Returns the result of an operational rule step assignment.
<i>Schedule</i>	Contains the scheduler information for a specific assignment.
<i>ScheduleCron</i>	Contains the scheduler information for a specific assignment in the form of a Crons spec.
<i>ScheduleDetails</i>	Returns the scheduler information for a specific assignment.
<i>ScheduleFrequency</i>	Returns a frequency object containing the scheduler frequency information for a specific assignment.
<i>ScheduleId</i>	Returns the unique BCM database identifier of the operational rule's schedule.
<i>ScheduleResult</i>	Returns a schedule object containing the scheduler information for a specific assignment.
<i>ScheduleValidity</i>	Returns a validity object containing the scheduler validity information for a specific assignment.
<i>StepAssignment</i>	Provides all information available for a specific step assignment.

Class	Description
<a href="#"><i>StepAssignmentArray</i></a>	Provides the list of all existing operational rule step assignments.
<a href="#"><i>StepAssignmentForUpdate</i></a>	Provides all the information required when updating an operational rule step assignment.
<a href="#"><i>StepAttrs</i></a>	Provides all the information required when adding a step to an operational rule.
<a href="#"><i>StepAttrsResult</i></a>	Provides all the information of a step that is assigned to an operational rule.
<a href="#"><i>StepClasses</i></a>	Provides the list of available step classes.
<a href="#"><i>StepDetails</i></a>	Provides details on a specific step assigned to a specific rule.
<a href="#"><i>StepList</i></a>	Provides the list of steps that are available in a specific class.
<a href="#"><i>StepParamAttrs</i></a>	Provides the values for the step parameters when assigning a step to an operational rule.
<a href="#"><i>StepParameters</i></a>	Provides the parameters and their values for a specific step.
<a href="#"><i>StepParameterValues</i></a>	Provides the information available for one specific step.
<a href="#"><i>StepParamValues</i></a>	Provides information about a specific step parameter and its value.
<a href="#"><i>Steps</i></a>	Provides information that is available for a specific step.
<a href="#"><i>User</i></a>	Provides user specific information for assignment.
<a href="#"><i>UserAssignment</i></a>	Provides all the information required when updating an operational rule user assignment.
<a href="#"><i>UserAssignmentAttrs</i></a>	Provides all available information about a specific operational rule user assignment.
<a href="#"><i>UserAssignmentAttrsArray</i></a>	Provides information about the assignment between an operational rule and a user.
<a href="#"><i>UserAssignmentAttrsObject</i></a>	Provides information about the assignment between an operational rule and a user.
<a href="#"><i>UserAssignmentAttrsPut</i></a>	Provides all necessary information to assign a specific user to an operational rule.
<a href="#"><i>UserAssignmentObj</i></a>	Provides all available information about a specific operational rule user assignment.
<a href="#"><i>UserGroup</i></a>	Provides user group specific information for assignment.

Class	Description
<i>UserGroupAssignment</i>	Provides all the information required when updating an operational rule user group assignment.
<i>UserGroupAssignmentAttrs</i>	Provides all available information about a specific operational rule user group assignment.
<i>UserGroupAssignmentAttrsArray</i>	Provides information about the assignment between an operational rule and a user group.
<i>UserGroupAssignmentAttrsObject</i>	Provides information about the assignment between an operational rule and a user group.
<i>UserGroupAssignmentObj</i>	Provides all available information about a specific operational rule user group assignment.

### AssignmentDetails

This class provides information about the assignment between an operational rule and a target.

Parameter	Type	Description
assignment	object[ <a href="#">AssignmentId</a> ]	One object for each assigned target.

### AssignmentId

This class provides the unique BCM database identifier of an assignment between an operational rule and its target.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between an operational rule and a target.

### AssignmentValuesResult

This class provides the values for the step parameters when assigning a step to an operational rule.

Parameter	Type	Description
Id	string	The unique BCM database identifier of the successful operational rule step assignment.

### Device (operational rule)

This class provides device specific information for assignment.

Parameter	Type	Description
Id	integer	The unique BCM database ID of the device. This value is returned by either the GET /devices or the GET /device operation.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
groupName	string	The name of the device group the device is a member of.

### DeviceAssignment

This class provides all the information required when updating an operational rule device assignment.

Parameter	Type	Description
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
transportMode	string	Defines in which mode the operational rule was be transferred to the targets, possible values are <code>Unknown</code> , until the operational rule is being transferred and then either <code>Multicast</code> or <code>Unicast</code> to indicate in which mode it was transferred.
scheduleId	string	The unique BCM database identifier of the operational rule's schedule. This value is returned by the <code>/oprule/rule/device/assignment/{assignmentId}</code> operation.
bypassTransferWindows	boolean	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
assignEnableTime	integer	Defines the condition on which the operational rule assignment will start towards the targets, this can either be <code>Immediately</code> or the date and time at which the activation is to be executed.
isActive	boolean	The overall status of the software distribution rule.

Parameter	Type	Description
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
specialCase	integer	.
wakeupDevices	boolean	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.
runAsCurrentUser	boolean	Set this value to true if the distribution is to be executed and installed on the local device as the logged user and not as LocalSystem.

### DeviceAssignmentAttrs

This class provides all available information about a specific operational rule - device assignment.



Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between the specific operational rule and a device. This value is returned by the <code>/oprule/rule/{ruleId}/device/assignments</code> operation.
device	object[ <a href="#">device</a> ]	One device object for each assigned device.
schedule	object[ <a href="#">schedule</a> ]	One schedule object for each device assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
transportMode	string	Defines in which mode the operational rule was be transferred to the targets, possible values are <code>Unknown</code> , until the operational rule is being transferred and then either <code>Multicast</code> or <code>Unicast</code> to indicate in which mode it was transferred.
assignEnableTime	integer	Defines the condition on which the operational rule assignment will start towards the targets, this can either be <code>Immediately</code> or the date and time at which the activation is to be executed.
isActive	boolean	The overall status of the software distribution rule.
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>

Parameter	Type	Description
wakeupDevices	boolean	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.
runAsCurrentUser	boolean	Set this value to true if the distribution is to be executed and installed on the local device as the logged user and not as LocalSystem.
executionError	array[ <a href="#">executionError</a> ]	One executionError array for each device assignment.

### DeviceAssignmentAttrsArray

This class provides a list of arrays containing information about the assignment between an operational rule and a device.

Parameter	Type	Description
assignments	array[ <a href="#">DeviceAssignmentAttrs</a> ]	This array provides all available information about a specific operational rule - device assignment.

### DeviceAssignmentAttrsObject

This class provides the list of operational rule device assignments, one array per assignment.

Parameter	Type	Description
assignments	object[ <a href="#">DeviceAssignmentObj</a> ]	This array provides all available information about a specific operational rule - device assignment.

### DeviceAssignmentObj

This class provides all available information about a specific operational rule - device assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between the specific operational rule and a device. This value is returned by the <code>/oprule/rule/{ruleId}/device/assignments</code> operation.
device	array[ <a href="#">device</a> ]	One device array for each assigned device.
schedule	array[ <a href="#">schedule</a> ]	One schedule array for each device assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
transportMode	string	Defines in which mode the operational rule was be transferred to the targets, possible values are <code>Unknown</code> , until the operational rule is being transferred and then either <code>Multicast</code> or <code>Unicast</code> to indicate in which mode it was transferred.
scheduleId	string	The unique BCM database identifier of the operational rule's schedule. This value is returned, depending on the assigned object - by either the <code>/oprule/rule/device/assignment/{assignmentId}</code> , <code>/oprule/rule/device/group/assignment/{assignmentId}</code> , <code>/oprule/rule/user/assignment/{assignmentId}</code> , or the <code>/oprule/rule/user/group/assignment/{assignmentId}</code> operation.
bypassTransferWindows	boolean	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
assignEnableTime	integer	Defines the condition on which the operational rule assignment will start towards the targets, this can either be <code>Immediately</code> or the date and time at which the activation is to be executed.
isActive	boolean	The overall status of the software distribution rule.

Parameter	Type	Description
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
specialCase	integer	.
wakeupDevices	boolean	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatus	boolean	Define if only the final status values, that is, <i>Executed</i> or <i>Failed</i> are to be uploaded (unchecked), or if each and every status that the operational rule execution is passing through is uploaded (checked). This option is only available if the corresponding system variable is activated.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.
runAsCurrentUser	boolean	Set this value to true if the distribution is to be executed and installed on the local device as the logged user and not as LocalSystem.
executionError	array[ <a href="#">executionError</a> ]	One executionError array for each device assignment.

## DeviceGroup

This class provides device group specific information for assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the group.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.

## DeviceGroupAssignment

This class provides all the information required when updating an operational rule device group assignment.

Parameter	Type	Description
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
scheduleId	string	The unique BCM database identifier of the operational rule's schedule. This value is returned by the <code>/oprule/rule/device/group/assignment/{assignmentId}</code> operation.
bypassTransferWindows	boolean	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
assignEnableTime	integer	Defines the condition on which the operational rule assignment will start towards the targets, this can either be <code>Immediately</code> or the date and time at which the activation is to be executed.
isActive	boolean	The overall status of the software distribution rule.

Parameter	Type	Description
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
specialCase	integer	.
wakeupDevices	boolean	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatus	boolean	Define if only the final status values, that is, <i>Executed</i> or <i>Failed</i> are to be uploaded (unchecked), or if each and every status that the operational rule execution is passing through is uploaded (checked). This option is only available if the corresponding system variable is activated.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.
runAsCurrentUser	boolean	Set this value to true if the distribution is to be executed and installed on the local device as the logged user and not as LocalSystem.

### DeviceGroupAssignmentAttrs

This class provides information about the assignment between an operational rule and a device group.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between the specific operational rule and a device. This value is returned by the <code>/oprule/rule/{ruleId}/device/assignments</code> operation.
device	array[ <a href="#">device</a> ]	One device array for each assigned device.
schedule	array[ <a href="#">schedule</a> ]	One schedule array for each device assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
transportMode	string	Defines in which mode the operational rule was be transferred to the targets, possible values are <code>Unknown</code> , until the operational rule is being transferred and then either <code>Multicast</code> or <code>Unicast</code> to indicate in which mode it was transferred.
bypassTransferWindows	boolean	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
assignEnableTime	integer	Defines the condition on which the operational rule assignment will start towards the targets, this can either be <code>Immediately</code> or the date and time at which the activation is to be executed.
isActive	boolean	The overall status of the software distribution rule.

Parameter	Type	Description
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
specialCase	integer	.
wakeupDevices	boolean	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
runAsCurrentUser	boolean	Set this value to true if the distribution is to be executed and installed on the local device as the logged user and not as LocalSystem.
executionError	array[ <a href="#">executionError</a> ]	One executionError array for each device assignment.

### DeviceGroupAssignmentAttrsArray

This class provides information about the assignment between an operational rule and a device group.

Parameter	Type	Description
assignments	array[ <a href="#">DeviceGroupAssignmentAttrs</a> ]	This array provides all available information about a specific operational rule - device group assignment.



## DeviceGroupAssignmentAttrsObject

This class provides information about the assignment between an operational rule and a device group.

Parameter	Type	Description
assignments	object[ <a href="#">DeviceGroupAssignmentObj</a> ]	This array provides all available information about a specific operational rule - device group assignment.

## DeviceGroupAssignmentObj

This class provides all available information about a specific operational rule - device group assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between the specific operational rule and a device group. This value is returned by the <code>GET /oprule/rule/{ruleId}/device/group/assignments</code> operation.
deviceGroup	object[ <a href="#">deviceGroup</a> ]	One device group object for each assigned group.
schedule	object[ <a href="#">schedule</a> ]	One schedule array for each device assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
bypassTransferWindows	boolean	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
assignEnableTime	integer	Defines the condition on which the operational rule assignment will start towards the targets, this can either be <code>Immediately</code> or the date and time at which the activation is to be executed.
isActive	boolean	The overall status of the software distribution rule.

Parameter	Type	Description
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
specialCase	integer	.
wakeupDevices	boolean	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatus	boolean	Define if only the final status values, that is, <i>Executed</i> or <i>Failed</i> are to be uploaded (unchecked), or if each and every status that the operational rule execution is passing through is uploaded (checked). This option is only available if the corresponding system variable is activated.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.
runAsCurrentUser	boolean	Set this value to true if the distribution is to be executed and installed on the local device as the logged user and not as LocalSystem.

## ExecutionError

This class provides all available error information of an operational rule execution on a specific target.

Parameter	Type	Description
errorStepNumber	integer	This number represents the number of the step which failed and thus caused the error.
errorType	string	This value shows of which type the occurred error is. Possible values are:<msgph>None</msgph>; for no error occurred,<msgph>Chilli Error </msgph>;if there was a problem with the Chilli execution script,<msgph>Module Error </msgph>;if the error was caused by any BCM module and<msgph>External Error</msgph>; if the error was not caused by a BCM module.
errorCode	integer	This number indicates the error number which may be the Chilli error code, a system error code, and so on.
errorText	string	This free text field displays a description of the error, if available.

### FolderAttrs

This class provides the information required to create or update an operational rules folder.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.

### FolderValuesResult

This object contains the unique BCM database identifier of the operational rules subfolder.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the operational rules subfolder.

### MyAppsIcons

This class provides the list of icons that are available for representing the rule in MyApps application kiosk.

Parameter	Type	Description
icons	List[string]	The names and paths of all MyApps icons that are available.

### OperationalRuleArray

This class returns the number of operational rules found in the specific location and one operational rule information array per rule. It is returned by the `GET /oprule/folder/{folderId}/rules` and `GET /oprule/rules` operations.

Parameter	Type	Description
total	integer	The total number of operational rules found in the specified location.
rules	array[ <a href="#">OperationalRuleValues</a> ]	One array for each operational rule that is found in this location.

### OperationalRuleAttrs

This class provides the information that is necessary to create a new operational rule. It is required by the `PUT /oprule/folder/{folderId}/rule` and `PUT /oprule/rule` operations.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
type	LIST	<p>The type of the operational rule, which may be either:</p> <ul style="list-style-type: none"> <li>• <b>Quick Link</b> if the rule is to be published to MyApps containing only a URL.</li> <li>• <b>Software Distribution</b> if a package is assigned to the operational rule for a software distribution, or</li> <li>• <b>Operational Rule</b> if it is any other type of action that is executed. An operational rule is always created with type <i>Operational Rule</i>. Once a package is assigned to it, its type is automatically set to <i>Software Distribution</i>.</li> </ul> <p>If the operational rule has been created through the assignment of a device under the <b>Packages</b> node the type is automatically set to <i>Software Distribution</i> and the name of the operational rule is <i>[pkgname.ext]</i>, <i>pkgname</i> being the name of the package to be distributed, <i>ext</i> indicating the type of the package.</p>
url	string	The URL
notes	string	This free text field can contain additional information concerning the selected object.

Parameter	Type	Description
kioskIcon	string	The icon that is displayed for the operational rule in MyApps.

### OperationalRuleAttrsForUpdate

This class provides the information that is necessary to update an existing operational rule. It is required by the `PUT /oprule/rule/{ruleId}` operation.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
url	string	The URL
notes	string	This free text field can contain additional information concerning the selected object.

### OperationalRuleFolderArray

This class provides the number of children of a specific location and information about all these children. It is returned by the `GET /oprule/subfolders` and `GET oprule/folder/{folderId}/subfolders` operations.

Parameter	Type	Description
total	integer	The total number of operational rules found in the specified location.
values	Array[ <a href="#">OperationalRuleFolderValues</a> ]	The available information for an operational rule folder.
errorCode	string	The error code returned by the operation if the operation failed or 0 for a successful operation.

### OperationalRuleFolderValues

This class provides the information available for an operational rules folder.

Parameter	Type	Description
folder	array[ <a href="#">OperationalRuleFolderValuesArray</a> ]	One array for each operational rule folder that is found in this location.

### OperationalRuleFolderValuesArray

This class provides the information available for an operational rules folder.

Parameter	Type	Description
Id	integer	The unique BCM database ID of the operational rule.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

### OperationalRuleValues

This class contains one array for each operational rule found in the specified location.

Parameter	Type	Description
rule	array[ <a href="#">OperationalRuleValuesArray</a> ]	One array for each operational rule that is found in this location.

### OperationalRuleValuesArray

This class returns the information that is available for an individual operational rule.

Parameter	Type	Description
Id	integer	The unique BCM database ID of the operational rule.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.

Parameter	Type	Description
type	string	<p>The type of the operational rule, which may be either:</p> <ul style="list-style-type: none"> <li>• <b>Quick Link</b> if the rule is to be published to MyApps containing only a URL.</li> <li>• <b>Software Distribution</b> if a package is assigned to the operational rule for a software distribution, or</li> <li>• <b>Operational Rule</b> if it is any other type of action that is executed. An operational rule is always created with type <i>Operational Rule</i>. Once a package is assigned to it, its type is automatically set to <i>Software Distribution</i>.</li> </ul> <p>If the operational rule has been created through the assignment of a device under the <b>Packages</b> node the type is automatically set to <i>Software Distribution</i> and the name of the operational rule is <i>[pkgname.ext]</i>, <i>pkgname</i> being the name of the package to be distributed, <i>ext</i> indicating the type of the package.</p>
deploymentFromExternalIntegration	string	<p>Defines if the operational rule to create is to be used with integrated solutions. This parameter is only available if at least one external integration is defined.</p> <ul style="list-style-type: none"> <li>• <b>Available in integrated solutions:</b> The rule is to be available in integrated solutions and may be scheduled from the BCM console.</li> <li>• <b>Additional workflow needed:</b> The rule is to be available in integrated solutions and can only be used from there, that is, it is not schedulable from the BCM console.</li> <li>• <b>Not available in integrated solutions:</b> The rule is only used by the BCM agents.</li> </ul>
kioskIcon	string	The icon that is displayed for the operational rule in MyApps.
url	string	The URL
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.

Parameter	Type	Description
updateDate	string	The date and time at which the object was last modified in the BCM database.

### OperationalRuleValuesResult

This class returns the unique id of the operational rule that was created. It is used by the `PutRuleResult` class.

Parameter	Type	Description
Id	integer	The id of the newly created operational rule.

### OperationResult

This class returns the result (error code) of the operation.

Parameter	Type	Description
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### OpRuleDependencies

This class returns the list of dependent operational rules on a specific rule.

Parameter	Type	Description
dependencies	object[ <a href="#">OpRuleDependencyDetails</a> ]	One dependency object for each dependent operational rule.

### OpRuleDependencyDetails

This class returns information about the dependent operational rules of a specific rule.

Parameter	Type	Description
rule	object[ <a href="#">OpRuleDependencyResult</a> ]	One rule dependency object for each dependent operational rule providing information this rule.
position	string	The position number of the dependend rule.

### OpRuleDependencyOrder

This class returns order of the dependent operational rules of a specific rule.

Parameter	Type	Description
dependency	object[ <a href="#">OpRuleDependencyPosition</a> ]	One dependency object for each dependent operational rule indicating the position of the dependent rule.



### OpRuleDependencyPosition

This class provides the position number of a specific dependent rule.

Parameter	Type	Description
position	integer	The position number of a specific dependent rule.

### OpRuleDependencyResult

This class returns information about the dependent operational rules of a specific rule.

Parameter	Type	Description
Id	integer	The unique BCM database ID of the operational rule.
name	integer	The name of the dependent operational rule.

### PackageAssignment

This class returns the list of packages assigned to a specific operational rule.

Parameter	Type	Description
packages	List[ <a href="#">PackageAssignmentDetails</a> ]	One packages object for each assigned package.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### PackageAssignmentDetails

This class returns information a specific package.

Parameter	Type	Description
Id	string	The unique BCM database identifier of the package.
name	string	The name of the package.
type	string	The type of the package, which can be either <code>CUSTOM</code> for a custom package, <code>MSI</code> for an MSI package, <code>RPM</code> for an RPM package, or <code>SNP</code> for a snapshot package.

### PutRuleFolderResult

This class returns the result of an operational rule folder creation operation.

Parameter	Type	Description
value	Object[ <a href="#">FolderValuesResult</a> ]	This object contains the unique BCM database identifier of the operational rules subfolder.

### PutRuleResult

This class returns the result of the create or update operation of the rule. It is returned by the `PUT /oprule/folder/{folderId}/rule` and `PUT /oprule/rule` operations.

Parameter	Type	Description
rule	<a href="#">OperationalRuleValuesResult</a>	This array returns the result and unique id of the operational rule that was created or updated.

### PutRuleStepAssignmentResult

This class provides the result of an operational rule step assignment.

Parameter	Type	Description
Id	string	The unique BCM database identifier of the step parameter. This value is returned by the <code>GET /oprule/step/{stepId}</code> operation.
value	string	The step parameter value.

### Schedule

This class contains the scheduler information for a specific assignment.

Parameter	Type	Description
name	string	The name of the scheduler.
validity	object[ <a href="#">ScheduleValidity</a> ]	A validity object containing the scheduler validity information for a specific assignment.
frequency	object[ <a href="#">ScheduleFrequency</a> ]	A frequency object containing the scheduler frequency information for a specific assignment.

### ScheduleCron

This class contains the scheduler information for a specific assignment in the form of a Cronspect.

The following general rules apply:

- Each set of ranges can be preceded by a % sign which will change the meaning from absolute to relative number. For instance if [seconds] equals 29 the timer will get fired each time the absolute time ends with a number of seconds equal to 29 (e.g., 11:43:29) whereas %20 means every 20 seconds every minute, that is, at 13:25:00, 13:25:20, 13:25:40, 13:26:00, and so on.

- Ranges are comma-separated lists. A range is made of a number eventually followed by a '-' sign and another number.
- The wildcard character asterisks (\*) can be used to indicate any value.

Parameter	Type	Description
second	string	The second value of the minute at which the execution is to start, it can vary from 0-59 with a maximum resolution of 5 seconds.
minute	string	The minute value of the hour at which the execution is to start, it can vary from 0-59.
hour	array[int]	The hour value at which the execution is to start, it can vary from 0-23.
day	array[int]	The number representing the days of the month that the object is to be executed, they can vary from 1-31.
month	array[int]	The number representing the months that the object is to be executed, they can vary from 1-12 (1 is January).
weekday	string	The number representing the week day that the object is to be executed, they can vary from 0-6 (0 is Sunday).

### ScheduleDetails

This class returns the scheduler information for a specific assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the operational rule's schedule.
name	string	The name of the operational rule the scheduler is assigned to.
validity	object[ <a href="#">ScheduleValidity</a> ]	A validity object containing the scheduler validity information for a specific assignment.
frequency	object[ <a href="#">ScheduleFrequency</a> ]	A frequency object containing the scheduler frequency information for a specific assignment.

### ScheduleFrequency

This class returns a frequency object containing the scheduler frequency information for a specific assignment.

Parameter	Type	Description
type	string	This value defines the frequency with which the operational rule is executed on the assigned target(s). The possible values are <code>startup</code> executes the rule the target device is booted, <code>sessionStartup</code> executes the rule when a the session is started on the target, <code>sessionClose</code> executes the rule when the session is closed on the target, <code>schedule</code> defines that the rule is executed according to a specific schedule defined via the following values and <code>cron</code> defines that the execution frequency is specified via a Cronspect.
scheduleType	string	This value defines the execution frequency. The possible values are <code>daily</code> for a daily execution, <code>dayOfWeek</code> which defines one or more specific days of the week for execution, or <code>dayOfMonth</code> in which case the rule is only executed on one or more specific days of each month.
scheduleDayOfWeek	array[int]	The days of the week on which the rule is executed, they can vary from 0-6 (0 is Sunday). This value is only applicable if <code>dayOfWeek</code> was previously selected as the <code>scheduleType</code> .
scheduleDayOfMonth	array[int]	The days of the month on which the rule is executed, they can vary from 1-31. This value is only applicable if <code>dayOfMonth</code> was previously selected as the <code>scheduleType</code> .
scheduleMonths	array[int]	The months in which the rule is executed according to the defined schedule, they can vary from 1-12.
period	string	The frequency at which the execution is launched. This can be <code>at</code> to launch only once per day, <code>1mn</code> to execute every minute of the previously selected frequency, <code>5/10/15/20/30mn</code> to execute every 5, 10, 15, 20, 30 and 60 minutes of the previously selected frequency.
periodAt	integer	The hour and minute values at which the rule is executed, in the form of <code>HH:MM</code> . This value is only applicable if the <code>period</code> is defined as <code>at</code> - Only once.

Parameter	Type	Description
periodFrom	array[int]	The hour and minute values from which on the rule is executed, in the form of HH:MM. This value is only applicable if the <code>period</code> is <i>not</i> defined as <code>at</code> - Only once .
periodTo	array[int]	The hour and minute values at which on the rule execution is stopped, in the form of HH:MM. This value is only applicable if the <code>period</code> is <i>not</i> defined as <code>at</code> - Only once .
cron	object[ScheduleCron]	The Cronspect schedule object containing the scheduler information in this format. The value is only applicable if the <code>type</code> is defined as <code>cron</code> .

### ScheduleId

This class returns the unique BCM database identifier of the operational rule's schedule.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the operational rule's schedule.

### ScheduleResult

This class returns a schedule object containing the scheduler information for a specific assignment.

Parameter	Type	Description
schedule	object[ScheduleDetails]	A schedule object containing the scheduler information for a specific assignment.

### ScheduleValidity

This class returns a validity object containing the scheduler validity information for a specific assignment.

Parameter	Type	Description
execution	string	Displays the condition on which the operational rule is executed on the assigned target(s). The possible values are <code>disabled</code> which indicates that the scheduler is currently disabled, <code>immediate</code> executes the rule immediately, <code>nextStartup</code> executes the rule at the next startup of the agent on the target and <code>at</code> executes the rule at a specific date and time.

Parameter	Type	Description
executionAt	integer	This value displays the date and time of execution in the form of a timestamp value. It is only applicable if the <code>at</code> option is previously selected as the <code>execution</code> .
termination	string	Displays the condition on which the operational rule is executed for the last time on the assigned target(s). The possible values are <code>never</code> in which case the rule is executed according to the defined schedule until another option is selected or it is unassigned, <code>runCount</code> executes the rule for the specified number of times and then terminates the execution cycle and <code>at</code> which terminates the execution cycle at a specific date and time.

### StepAssignmentArray

This class provides the list of all existing operational rule step assignments.

Parameter	Type	Description
assignments	Array[ <a href="#">StepAssignment</a> ]	This array provides one object per assignment containing all information available for the specific step assignment.

### StepAssignmentForUpdate

This class provides all the information required when updating an operational rule step assignment.

Parameter	Type	Description
position	integer	The position at which the step is to be placed in the list of steps.
params	Array[ <a href="#">StepParamAttrs</a> ]	This array provides all values for the parameters of the step. If only the position is to be modified, it is not required.

### StepAttrs

This class provides all the information required when adding a step to an operational rule.

Parameter	Type	Description
stopCondition	LIST	Defines if the operational rule is to continue in case of error of a step. The possible values are <code>_DB_STEPACTION_CONTINUE_</code> : Continue with the execution of the operational rule, no matter if the current step was executed successfully or if it failed. <code>_DB_STEPACTION_FAILED_</code> : Stop the execution of the operational rule if this step failed and exit with a failed execution condition. or <code>_DB_STEPACTION_SUCCEED_</code> : Stop the execution of the operational rule if this step is successful and exit with a successful execution condition..
verificationCondition	LIST	Defines if the operational rule is to continue the verification phase of a step if it fails. The possible values are <code>_DB_STEPACTION_VERIFLOOPONERROR_</code> : Continue with the verification phase as long as the current step fails., <code>_DB_STEPACTION_VERIFLOOPONSUCCESS_</code> : Continue with the verification phase as long as the current step succeeds., <code>_DB_STEPACTION_VERIFNONE_</code> : Do not execute any verification on the step. This option should be used if the step should generate an alert., <code>_DB_STEPACTION_VERIFSTOPONERROR_</code> : This parameter verifies the target device for a specific condition. If this verification fails, that is, the condition is not fulfilled, the operational rule is considered as already executed, and its status is updated to <i>Executed</i> . Any further steps defined in this rule is ignored. or <code>_DB_STEPACTION_VERIFSTOPONSUCCESS_</code> : This parameter verifies the target device for a specific condition. If this verification is successful, that is, the condition is fulfilled, the operational rule is considered as already executed, and its status is updated to <i>Executed</i> . Any further steps defined in this rule is ignored..
notes	string	This free text field can contain additional information concerning the selected object.
params	Array[ <a href="#">StepParamAttrs</a> ]	This array provides all values for the parameters of the step.

### StepAttrsResult

This class provides all the information of a step that is assigned to an operational rule.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the step parameter. This value is returned by the <code>GET /oprule/step/{stepId}</code> operation.
notes	string	This free text field can contain additional information concerning the selected object.
params	Array[ <a href="#">StepParamAttrs</a> ]	This array provides all values for the parameters of the step.

### StepClasses

This class provides the list of available step classes.

Parameter	Type	Description
classes	List[string]	The names of all classes that are available.

### StepDetails

This class provides details on a specific step assigned to a specific rule.

Parameter	Type	Description
step	object[ <a href="#">StepParameters</a> ]	One object for each step parameter.

### StepList

This class provides the list of steps that are available in the specified class.

Parameter	Type	Description
steps	array[ <a href="#">Steps</a> ]	One array for each step that is part of the class.

### StepParamAttrs

This class provides the values for the step parameters when assigning a step to an operational rule.

Parameter	Type	Description
Id	string	The unique BCM database identifier of the step parameter. This value is returned by the <code>GET /oprule/step/{stepId}</code> operation.



Parameter	Type	Description
value	string	The step parameter value.

### StepParameters

This class provides the parameters and their values for a specific step.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the step parameter. This value is returned by the GET /oprule/step/{stepId} operation.
params	Array[ <a href="#">StepParameterValues</a> ]	One array for each step parameter.

### StepParameterValues

This class provides a specific step parameters and its value.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the step parameter. This value is returned by the GET /oprule/step/{stepId} operation.
label	integer	The keyword which is used to provide the name of the parameter in the selected locale.
name	string	The internal name of the parameter.
type	string	The data type of the step parameter.

### StepParamValues

This class provides information about a specific step parameter and its value.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the step parameter. This value is returned by the GET /oprule/step/{stepId} operation.
label	string	The keyword which is used to provide the name of the parameter in the selected locale.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
type	string	The data type of the step parameter.
value	string	The value of the step parameter, that it currently has or is to be assigned.

## Steps

This class provides information that is available for a specific step.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the operational rule step. This value is returned by the <code>GET /oprule/steps</code> operation.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
class	string	The name of the class the step is a member of.
notes	string	This free text field can contain additional information concerning the selected object.

## User

This class provides user specific information for assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the user. User IDs are returned by the <code>GET /user/users</code> or <code>GET /user/group/{groupId}/users</code> operation.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
group	array[ <a href="#">UserGroup</a> ]	One UserGroup array for each group the user is a member of.

## UserAssignment

This class provides all the information required when updating an operational rule user assignment.

Parameter	Type	Description
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
scheduleId	string	The unique BCM database identifier of the operational rule's schedule. This value is returned by the <code>/oprule/rule/user/assignment/{assignmentId}</code> operation.
installAsUser	boolean	Set this value to true if the object is to be executed and installed on the local device as the logged user instead of with the system account.
notOnlyPrimaryUser	boolean	Set this value to true if the assignments are applicable to the secondary users as well as the primary user. If set to false the assignments will only be carried out for the primary user.
userAssignPolicy	string	<p>Defines the type of policy to use for the user assignments:</p> <ul style="list-style-type: none"> <li>• <b>Assign at User Login:</b> the assignment of the operational rules is carried out when the user logs on to a device.</li> <li>• <b>Assign at User Logout:</b> the assignment is carried out when the user logs off.</li> <li>• <b>Assign Immediately:</b> the assignment is carried out immediately after defining the assignment.</li> <li>• <b>Assign with Default Assignment Date:</b> the assignment is carried out according to the defined default schedule. This option is only available if the <b>Deploy to devices linked to users</b> option is previously selected.</li> </ul>

Parameter	Type	Description
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
bypassTransferWindows	boolean	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
wakeupDevices	boolean	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.

### UserAssignmentAttrs

This class provides all available information about a specific operational rule - user assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between the specific operational rule and a specific user. This value is returned by the <code>GET /oprule/rule/{ruleId}/user/assignments</code> operation.

Parameter	Type	Description
user	object[user]	One user object for each assigned user.
schedule	object[schedule]	One schedule object for each user assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
CreateTime	string	The date and time the file was created on the originating device. This is the same information as that of the <b>Name</b> field part but in the default date and time format defined in the user preferences.
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
installAsUser	string	Set this value to true if the object is to be executed and installed on the local device as the logged user instead of with the system account.

Parameter	Type	Description
userAssignPolicy	string	<p>Defines the type of policy to use for the user assignments:</p> <ul style="list-style-type: none"> <li>• <b>Assign at User Login:</b> the assignment of the operational rules is carried out when the user logs on to a device.</li> <li>• <b>Assign at User Logout:</b> the assignment is carried out when the user logs off.</li> <li>• <b>Assign Immediately:</b> the assignment is carried out immediately after defining the assignment.</li> <li>• <b>Assign with Default Assignment Date:</b> the assignment is carried out according to the defined default schedule. This option is only available if the <b>Deploy to devices linked to users</b> option is previously selected.</li> </ul>

### UserAssignmentAttrsArray

This class provides information about the assignment between an operational rule and a user.

Parameter	Type	Description
assignments	array[ <a href="#">UserAssignmentAttrs</a> ]	This array provides all available information about a specific operational rule - user assignment.

### UserAssignmentAttrsObject

This class provides information about the assignment between an operational rule and a user.

Parameter	Type	Description
assignments	object[ <a href="#">UserAssignmentObj</a> ]	This object provides all available information about a specific operational rule - user assignment.

### UserAssignmentAttrsPut

This class provides all necessary information to assign a specific user to an operational rule.

Parameter	Type	Description
installAsUser	boolean	Set this value to true if the object is to be executed and installed on the local device as the logged user instead of with the system account.

Parameter	Type	Description
modeMyApps	boolean	Set this value to true if the operational rule is to be assigned to the user via MyApps instead of being assigned to the user and the devices of which is he a user.
notOnlyPrimaryUser	boolean	Set this value to true if the assignments are applicable to the secondary users as well as the primary user. If set to false the assignments will only be carried out for the primary user.
userAssignPolicy	LIST	<p>Defines the type of policy to use for the user assignments:</p> <ul style="list-style-type: none"> <li>• <b>_ASSIGN_USERLOGIN_</b>: the assignment of the operational rules is carried out when the user logs on to a device.</li> <li>• <b>_ASSIGN_USERLOGOUT_</b>: the assignment is carried out when the user logs off.</li> <li>• <b>_ASSIGN_IMMEDIATELY_</b>: the assignment is carried out immediately after defining the assignment.</li> <li>• <b>_ASSIGN_DEFAULTSCHEDULE_</b>: the assignment is carried out according to the defined default schedule. This option is only available if the <b>Deploy to devices linked to users</b> option is previously selected.</li> </ul>

### UserAssignmentObj

This class provides all available information about a specific operational rule - user assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between the specific operational rule and a specific user. This value is returned by the <code>GET /oprule/rule/{ruleId}/user/assignments</code> operation.
user	object[user]	One device object for each assigned user.
schedule	object[schedule]	One schedule object for each user assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.

Parameter	Type	Description
installAsUser	string	Set this value to true if the object is to be executed and installed on the local device as the logged user instead of with the system account.
notOnlyPrimaryUser	string	Set this value to true if the assignments are applicable to the secondary users as well as the primary user. If set to false the assignments will only be carried out for the primary user.
userAssignPolicy	string	<p>Defines the type of policy to use for the user assignments:</p> <ul style="list-style-type: none"> <li>• <b>Assign at User Login:</b> the assignment of the operational rules is carried out when the user logs on to a device.</li> <li>• <b>Assign at User Logout:</b> the assignment is carried out when the user logs off.</li> <li>• <b>Assign Immediately:</b> the assignment is carried out immediately after defining the assignment.</li> <li>• <b>Assign with Default Assignment Date:</b> the assignment is carried out according to the defined default schedule. This option is only available if the <b>Deploy to devices linked to users</b> option is previously selected.</li> </ul>
networkInstall	string	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
bypassTransferWindows	string	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.



Parameter	Type	Description
wakeupDevices	string	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatus	boolean	Define if only the final status values, that is, <i>Executed</i> or <i>Failed</i> are to be uploaded (unchecked), or if each and every status that the operational rule execution is passing through is uploaded (checked). This option is only available if the corresponding system variable is activated.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.

### UserGroup

This class provides user group specific information for assignment.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the user group. User group IDs are returned by the GET /user/subgroups or GET /user/group/{groupId}/subgroups operation.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.

### UserGroupAssignment

This class provides all the information required when updating an operational rule user group assignment.

Parameter	Type	Description
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.

Parameter	Type	Description
scheduleId	string	The unique BCM database identifier of the operational rule's schedule. This value is returned by the <code>/oprule/rule/user/group/assignment/{assignmentId}</code> operation.
installAsUser	string	Set this value to true if the object is to be executed and installed on the local device as the logged user instead of with the system account.
notOnlyPrimaryUser	string	Set this value to true if the assignments are applicable to the secondary users as well as the primary user. If set to false the assignments will only be carried out for the primary user.
userAssignPolicy	string	<p>Defines the type of policy to use for the user assignments:</p> <ul style="list-style-type: none"> <li>• <b>Assign at User Login:</b> the assignment of the operational rules is carried out when the user logs on to a device.</li> <li>• <b>Assign at User Logout:</b> the assignment is carried out when the user logs off.</li> <li>• <b>Assign Immediately:</b> the assignment is carried out immediately after defining the assignment.</li> <li>• <b>Assign with Default Assignment Date:</b> the assignment is carried out according to the defined default schedule. This option is only available if the <b>Deploy to devices linked to users</b> option is previously selected.</li> </ul>

Parameter	Type	Description
networkInstall	string	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
bypassTransferWindows	string	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
wakeupDevices	string	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	string	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatusEveryExec	string	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.

### UserGroupAssignmentAttrs

This class provides all available information about a specific operational rule - user group assignment.

Parameter	Type	Description
Id	integer	<p>The unique BCM database identifier of the assignment between the specific operational rule and a specific user group. This value is returned by the <code>GET /oprule/rule/{ruleId}/user/group/assignments</code> operation.</p>

Parameter	Type	Description
userGroup	object[userGroup]	One user group object for each assigned user group.
schedule	object[schedule]	One schedule object for each user assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
CreateTime	string	The date and time the file was created on the originating device. This is the same information as that of the <b>Name</b> field part but in the default date and time format defined in the user preferences.
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
installAsUser	string	Set this value to true if the object is to be executed and installed on the local device as the logged user instead of with the system account.

Parameter	Type	Description
userAssignPolicy	string	<p>Defines the type of policy to use for the user assignments:</p> <ul style="list-style-type: none"> <li>• <b>Assign at User Login:</b> the assignment of the operational rules is carried out when the user logs on to a device.</li> <li>• <b>Assign at User Logout:</b> the assignment is carried out when the user logs off.</li> <li>• <b>Assign Immediately:</b> the assignment is carried out immediately after defining the assignment.</li> <li>• <b>Assign with Default Assignment Date:</b> the assignment is carried out according to the defined default schedule. This option is only available if the <b>Deploy to devices linked to users</b> option is previously selected.</li> </ul>

### UserGroupAssignmentAttrsArray

This class provides information about the assignment between an operational rule and a user group.

Parameter	Type	Description
assignments	array[ <a href="#">UserGroupAssignmentAttrs</a> ]	This array provides all available information about a specific operational rule - user group assignment.

### UserGroupAssignmentAttrsObject

This class provides information about the assignment between an operational rule and a user group.

Parameter	Type	Description
assignments	object[ <a href="#">UserGroupAssignmentObj</a> ]	This object provides all available information about a specific operational rule - user group assignment.

### UserGroupAssignmentObj

This class provides all available information about a specific operational rule - user group assignment.

Parameter	Type	Description
Id	integer	<p>The unique BCM database identifier of the assignment between the specific operational rule and a specific user group. This value is returned by the <code>GET /oprule/rule/{ruleId}/user/group/assignments</code> operation.</p>

Parameter	Type	Description
userGroup	object[userGroup]	One device object for each assigned user group.
schedule	object[schedule]	One schedule object for each user group assignment.
status	string	The overall status of the assigned object, for example if the operational rule has been downloaded and is ready to be executed or has already been successfully installed.
installAsUser	boolean	Set this value to true if the object is to be executed and installed on the local device as the logged user instead of with the system account.
notOnlyPrimaryUser	boolean	Set this value to true if the assignments are applicable to the secondary users as well as the primary user. If set to false the assignments will only be carried out for the primary user.
userAssignPolicy	string	<p>Defines the type of policy to use for the user assignments:</p> <ul style="list-style-type: none"> <li>• <b>Assign at User Login:</b> the assignment of the operational rules is carried out when the user logs on to a device.</li> <li>• <b>Assign at User Logout:</b> the assignment is carried out when the user logs off.</li> <li>• <b>Assign Immediately:</b> the assignment is carried out immediately after defining the assignment.</li> <li>• <b>Assign with Default Assignment Date:</b> the assignment is carried out according to the defined default schedule. This option is only available if the <b>Deploy to devices linked to users</b> option is previously selected.</li> </ul>

Parameter	Type	Description
networkInstall	boolean	<p>Defines the type of installation of the package:</p> <ul style="list-style-type: none"> <li>• <b>Administrative:</b> Select this option if the package is to be installed in the classic MSI administrative installation mode. This option is only applicable to MSI packages.</li> <li>• <b>Network:</b> Select this option if you want to extract the MSI/custom package created with BCM Deploy into a specific network path from which the target devices download.</li> <li>• <b>Normal:</b> Select this option if the package is to be simply downloaded by the target devices and installed.</li> </ul> <p>This parameter is only available if the corresponding system variable is activated.</p>
bypassTransferWindows	string	Set this value to true if the distribution assignment is to be sent directly, ignoring any transfer window specifications which exist for the targets.
wakeupDevices	string	Set this value to true if the agent is to wake up any devices which are currently switched off, to immediately execute the assignment instead of waiting for the next startup to do so.
executeWhileFails	boolean	Defines if the operational rule/package is to be executed until its execution finally succeeds, that is, the final status <i>Executed</i> is uploaded.
uploadStatus	boolean	Define if only the final status values, that is, <i>Executed</i> or <i>Failed</i> are to be uploaded (unchecked), or if each and every status that the operational rule execution is passing through is uploaded (checked). This option is only available if the corresponding system variable is activated.
uploadStatusEveryExec	boolean	Defines that the status value is uploaded after every execution of the rule, even if it has not changed.

## Reports

The reports object type allows you to execute operations on a collection of reports.

*Return Classes*  
*Possible reports operations*

Operations on Reports

The following operations are available on the reports object:

GET

The following get operations are available:

Action	Description
<a href="#">/reports/folders</a>	Get report folders under root node.
<a href="#">/reports/folder/{folderId}</a>	Get report folder subfolders.
<a href="#">/reports/reports</a>	List reports at root level.
<a href="#">/reports/folder/{folderId}/reports</a>	This operation returns the list of reports located below a specific report folder.
<a href="#">/reports/public</a>	Return the list of all public reports.

GET Operations

/reports/folders

This operation returns the report folders located directly at the root level of the main Reports node.

Syntax

GET /reports/folders?  
brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowReports&offset=OffsetStart&count=Entries

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the package folders or also other available information, true for IDs only, false for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder (name), the date at which the folder was created (createDate) or the date at which it was last modified (updateDate).



Name	Type	Mand.	Data Type	Description
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the packages are listed in the return class as well, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ReportFolderArray](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

## Example

This example returns the list of all folders that are located directly below the **Reports** top node).

### Request URL

```
https://localhost:1610/api/1/reports/folders?
brief=false&sort=name&asc=true&hierarchy=false&count=100
```

### Response Body

```
{
  "folders": [
    {
      "id": 162,
      "name": "Application Usage",
      "createDate": 1361527322,
      "updateDate": 1361527336
    },
    {
      "id": 164,
      "name": "Compliance",
      "createDate": 1361527322,
      "updateDate": 1361527337
    }
  ],
}
```

```

{
  "id": 160,
  "name": "Distribution Statistics",
  "createDate": 1361527322,
  "updateDate": 1361527334
},
{
  "id": 181,
  "name": "Financial Asset Management",
  "createDate": 1361527328,
  "updateDate": 1361527357
},
{
  "id": 220,
  "name": "Out of the Box",
  "createDate": 1373360811,
  "updateDate": 1373360817
},
{
  "id": 165,
  "name": "Patches",
  "createDate": 1361527322,
  "updateDate": 1361527337
},
{
  "id": 161,
  "name": "Power Management",
  "createDate": 1361527322,
  "updateDate": 1361527335
},
{
  "id": 180,
  "name": "Security Products Management",
  "createDate": 1361527328,
  "updateDate": 1361527357
},
{
  "id": 167,
  "name": "Software Licence Management",
  "createDate": 1361527322,
  "updateDate": 1361527338
},
{
  "id": 166,
  "name": "Task Management",
  "createDate": 1361527322,
  "updateDate": 1361527337
},
{
  "id": 168,
  "name": "Virtualisation",
  "createDate": 1361527322,
  "updateDate": 1361527338
}
]
}

```

### **/reports/folder/{folderId}**

This operation returns the report folders that are members of a specific report folder.

### **Syntax**

GET /reports/folder/{folderId}?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowReports&offset=OffsetStart&count=Entries

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the report folder for which the hierarchy is to be provided. This value was either provided by a prior execution of this operation or by the <code>GET /reports/folders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the package folders or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).

Name	Type	Mand.	Data Type	Description
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the packages are listed in the return class as well, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ReportFolderArray](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

## Example

This example finds all children, folders and reports of a report folder with the ID 164.

### Request URL

```
https://localhost:1611/api/1/reports/folder/164?
brief=false&sort=name&asc=true&hierarchy=true&count=100
```

### Response Body

```
{
  "folders": [],
  "reports": [
    {
      "id": 1061,
      "name": "Compliance by Criteria",
      "reportTitle": "Compliance by Criteria",
      "reportType": "_DB_TEMPLATE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527337
    },
    {
      "id": 1063,
      "name": "Compliance by Device",
```

```

    "reportTitle": "Compliance by Device",
    "reportType": "_DB_TEMPLATE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527337
  },
  {
    "id": 1062,
    "name": "Compliance by Device Group",
    "reportTitle": "Compliance by Device Group",
    "reportType": "_DB_TEMPLATE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527337
  },
  {
    "id": 1064,
    "name": "Executive Compliance Summary",
    "reportTitle": "Executive Compliance Summary",
    "reportType": "_DB_TEMPLATE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527337
  },
  {
    "id": 1060,
    "name": "Non-compliance Reasons by Device",
    "reportTitle": "Non-compliance Reasons by Device",
    "reportType": "_DB_TEMPLATE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527337
  }
]
}

```

## /reports/reports

This operation returns the reports located directly at the root level of the main Reports node.

## Syntax

GET /reports/reports?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the package folders or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).

Name	Type	Mand.	Data Type	Description
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ReportArray](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

## Example

This example lists all reports directly located below the **Reports** top node).

### Request URL

```
https://localhost:1610/api/1/reports/reports?brief=false&sort=name&asc=true&count=100
```

### Response Body

```
{
  "reports": [
    {
      "id": 1014,
      "name": "Active Directory Devices",
      "reportTitle": "Active Directory Devices",
      "reportType": "_DB_STYLE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527333
    },
    {
      "id": 1013,
      "name": "Auto-discovered devices",
      "reportTitle": "Auto-discovered devices",
      "reportType": "_DB_STYLE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527333
    },
    {
      "id": 1046,
```

```

    "name": "Configuration Summary",
    "reportTitle": "Configuration Summary",
    "reportType": "_DB_TEMPLATE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527336
  },
  {
    "id": 1079,
    "name": "Deprecated Devices",
    "reportTitle": "Deprecated Devices",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1364473283
  },
  {
    "id": 1000,
    "name": "Device Users",
    "reportTitle": "Device Users",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527321,
    "updateDate": 1366884288
  },
  {
    "id": 1001,
    "name": "Disks",
    "reportTitle": "Disks",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527321,
    "updateDate": 1361527333
  },
  {
    "id": 1008,
    "name": "<term conref = \"../variables/variables.xml#id10B8A000W5Z/term_FullName\"/> Agent Versions",
    "reportTitle": "<term conref = \"../variables/variables.xml#id10B8A000W5Z/term_FullName\"/> Agent Versions",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527333
  },
  {
    "id": 1002,
    "name": "Global Device List",
    "reportTitle": "Global Device List",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527333
  },
  {
    "id": 1003,
    "name": "Global Software List",
    "reportTitle": "Global Software List",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527333
  },
  {
    "id": 1004,
    "name": "Hardware Summary List",
    "reportTitle": "Hardware Summary List",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527333
  },
  {
    "id": 1005,
    "name": "Memory Breakdown",
    "reportTitle": "Memory Breakdown",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,

```

```

    "updateDate": 1361527333
  },
  {
    "id": 1006,
    "name": "Network List",
    "reportTitle": "Network List",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527333
  },
  {
    "id": 1226,
    "name": "New Report",
    "reportTitle": "",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1375179985,
    "updateDate": 1375179988
  },
  {
    "id": 1007,
    "name": "Operating System Breakdown",
    "reportTitle": "Operating System Breakdown",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527333
  },
  {
    "id": 1009,
    "name": "Processor Breakdown",
    "reportTitle": "Processor Breakdown",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527333
  },
  {
    "id": 1080,
    "name": "Scanned Devices (Inventory)",
    "reportTitle": "Scanned Devices (Inventory)",
    "reportType": "_DB_STYLE_BASED_",
    "createDate": 1361527322,
    "updateDate": 1361527338
  }
]
}

```

### **/reports/folder/{folderId}/reports**

This operation returns the list of reports located below a specific report folder.

### **Syntax**

GET /reports/folder/{folderId}/reports{folderId}?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

### **Parameters**

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the report folder for which the hierarchy is to be provided. This value was either provided by a prior execution of this operation or by the <code>GET /reports/folders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the package folders or also other available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ReportArray](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

## Example

This example finds all reports of a report folder with the ID 164.

### Request URL

<https://localhost:1611/api/1/reports/folder/164/reports?brief=false&sort=name&asc=true&count=100>

### Response Body

```
{
  "reports": [
    {
      "id": 1061,
      "name": "Compliance by Criteria",
      "reportTitle": "Compliance by Criteria",
      "reportType": "_DB_TEMPLATE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527337
    },
    {
      "id": 1063,
      "name": "Compliance by Device",
      "reportTitle": "Compliance by Device",
      "reportType": "_DB_TEMPLATE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527337
    },
    {
      "id": 1062,
      "name": "Compliance by Device Group",
      "reportTitle": "Compliance by Device Group",
      "reportType": "_DB_TEMPLATE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527337
    }
  ]
}
```

```

    },
    {
      "id": 1064,
      "name": "Executive Compliance Summary",
      "reportTitle": "Executive Compliance Summary",
      "reportType": "_DB_TEMPLATE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527337
    },
    {
      "id": 1060,
      "name": "Non-compliance Reasons by Device",
      "reportTitle": "Non-compliance Reasons by Device",
      "reportType": "_DB_TEMPLATE_BASED_",
      "createDate": 1361527322,
      "updateDate": 1361527337
    }
  ]
}

```

### **/reports/public**

This operation returns the list of all reports generated within a specific timeframe that are publicly available.

### **Syntax**

GET /reports/public?DateFrom=StartDate&DateTo=EndDate

### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
DateFrom	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The earliest possible date that a public report to be found can were created on. The date must be entered in the following format: MM/DD/YYYY.

Name	Type	Mand.	Data Type	Description
DateTo	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The last possible date that a public report to be found can were created on. The date must be entered in the following format: MM/DD/YYYY.

## Returns

Returns the class [ReportsPublic](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

## Example

This example finds all public reports between the 1st of January and May 15, 2013.

### Request URL

```
https://localhost:1611/api/1/reports/public?DateFrom=01%2F01%2F2013&DateTo=05%2F15%2F2013
```

### Response Body

```
{
  "Version": "1.0",
  "DateFrom": "01/01/2013",
  "DateTo": "05/13/2013",
  "HttpProtocolRequestInfo": {
    "HttpProtocolRequestUser": "admin"
  },
  "Values": [
    {
      "ObjectId": 1002,
      "ReportName": "Device Users",
      "ReportTitle": "Device Users",
      "ReportDate": "25 April 2013 12:04:57",
      "GroupName": "",
      "XMLStatus": "_STATUS_NOTREQUESTED_",
      "HTMLStatus": "_STATUS_AVAILABLE_",
      "PDFStatus": "_STATUS_NOTREQUESTED_",
      "ReportFile": "Device_Users_2013_04_25_12_04_57.html",
      "Path": "ID_1000/2D29165E66B8CCE3AEFF3CF5AB563C67/Device_Users_2013_04_25_12_04_57.html"
    },
    {
      "ObjectId": 1004,
      "ReportName": "Device Users",
      "ReportTitle": "Device Users",
      "ReportDate": "03 May 2013 15:25:53",
      "GroupName": "",
      "XMLStatus": "_STATUS_NOTREQUESTED_",
      "HTMLStatus": "_STATUS_AVAILABLE_",
      "PDFStatus": "_STATUS_NOTREQUESTED_",
      "ReportFile": "Device_Users_2013_05_03_15_25_53.html",
      "Path": "ID_1000/C5E8AE2453E45E98BEFA08718AE1FF3F/Device_Users_2013_05_03_15_25_53.html"
    }
  ],
  "ErrorCode": 0
}
```

In this example two public reports are available, both in HTML format, for the requested period of time.

## Reports Return Classes

The following classes can be returned by the different reports actions:

Class	Description
<a href="#"><i>ReportArray</i></a>	Get list of reports of a folder.
<a href="#"><i>ReportFolderArray</i></a>	Get list of report folders of a folder.
<a href="#"><i>ReportFolderInfo</i></a>	Return information about one specific report folder.
<a href="#"><i>ReportInfo</i></a>	Return information about one specific report.
<a href="#"><i>ReportsPublic</i></a>	Return the list of all public reports.

Class	Description
<a href="#"><i>ReportsPublicValues</i></a>	Data of a specific public report.

### ReportArray

This class returns the list of all reports, that are children of a specific report folder including the main Reports node.

Parameter	Type	Description
reports	Array[ <a href="#"><i>ReportInfo</i></a> ]	An array of type <code>ReportFolderInfo</code> listing all folders and reports located under the requested location.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ReportFolderArray

This class returns the list of all folders and, if requested, all reports, that are children of a specific report folder including the main Reports node.

Parameter	Type	Description
folders	Array[ <a href="#"><i>ReportFolderInfo</i></a> ]	The array of type <code>ReportFolderArray</code> listing all folders located under the top node.
reports	Array[ <a href="#"><i>ReportArray</i></a> ]	The array of type <code>ReportFolderArray</code> listing all folders located under the top node.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ReportFolderInfo

This class returns available information about one report folder found in the requested location.

Parameter	Type	Description
objectId	integer	The unique BCM identifier of the report folder.
name	string	The name of the object type.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

## ReportInfo

This class returns available information about one report found in the requested location.

Parameter	Type	Description
objectId	integer	The unique BCM identifier of the report.
name	string	The name of the object type.
reportTitle	string	The title of report.
reportType	string	The type of the report, that is, if it is style or template based.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

## ReportsPublic

This class returns the list of all generated public reports.

Parameter	Type	Description
Values	List[ <a href="#">ReportsPublicValues</a> ]	An array of type <code>ReportPublicValues</code> listing all available public reports for the requested timeframe.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

## ReportsPublicValues

This class returns the data for a specific public report.

Parameter	Type	Description
objectId	integer	The unique BCM database identifier of the object.
ReportName	string	The user friendly name of the report.
ReportTitle	string	The title of the report.
ReportDate	string	The date and time at which the report was generated and made publicly available.
GroupName	string	The name or names of the object the public report is assigned to.



Parameter	Type	Description
XMLStatus	string	Indicates if the public report is available in XML format, possible values are <code>_STATUS_AVAILABLE_</code> for available or <code>_STATUS_NOTREQUESTED_</code> if the report was not generated in this format.
HTMLStatus	string	Indicates if the public report is available in HTML format, possible values are <code>_STATUS_AVAILABLE_</code> for available or <code>_STATUS_NOTREQUESTED_</code> if the report was not generated in this format.
PDFStatus	string	Indicates if the public report is available in PDF format, possible values are <code>_STATUS_AVAILABLE_</code> for available or <code>_STATUS_NOTREQUESTED_</code> if the report was not generated in this format.
ReportFile	string	The filename of the report.
path	string	The path to the file of the report including the filename.

## Report

The report object type allows you to execute operations on specific reports.

[Report operations](#)

[Return Classes](#)

## Report Operations

The following operations are available for the report object:

### GET

The following get operations are available:

Action	Description
<a href="#">/report/{reportId}/generate</a>	Generate specific public report.
<a href="#">/report/{reportId}/load</a>	Load public report from archive.

GET Operations

/report/{reportId}/generate

This operation generates a specific public report.

Syntax

GET /report/{reportId}/generate

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
reportId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the report. This value is provided by the GET /reports/public operation.

Returns

Returns the class *ReportGenerate*.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

## Example

The following example successfully generates the report (ID 1009) as provided by operation `/reports/folder/{folderId}`.

### Request URL

```
https://localhost:1611/api/1/report/1009/generate
```

### Response Body

```
{
  "Value": [
    1006
  ],
  "ErrorCode": 0
}
```

## /report/{reportId}/load

This operation loads a specific, already generated public report from the archive and displays it on the screen.

## Syntax

```
GET /report/{reportId}/load
```

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
reportId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the report. This value is returned by the GET /reports/public operation.

## Returns

Returns the class [ReportLoad](#).

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
12	Access denied. Please verify your access rights.

## Example

The following example loads the generated public report with the ID 1004 as provided by operation `/reports/public`.

### Request URL

```
https://localhost:1611/api/1/report/1004/load
```

### Response Body

```
{
  "DeviceHttpPort": 1610,
  "Values": "D:/Program Files/BMC Software/Panoramix/data/Vision64Database\\reports\\ID_1000\\C5E8AE2453E45E98BEFA08718AE1FF3F\\Device_Users_2013_05_03_15_25_53.html",
  "GroupId": 0,
  "ObjectId": 1004,
  "CreateTime": "2013-05-03 13:25:53.000",
  "ErrorCode": 0
}
```

## Report Return Classes

The following classes can be returned by the different report actions:

Class	Description
<a href="#"><i>ReportGenerate</i></a>	Return the result of a report generation.
<a href="#"><i>ReportLoad</i></a>	Data of an archived public report.
<a href="#"><i>ReportLoadReportData</i></a>	Returns the report specific data of a style-based report.
<a href="#"><i>ReportLoadSubReportData</i></a>	Returns the subreport data.

### ReportGenerate

This class returns if the generation of a specific public report was successful.

Parameter	Type	Description
Value	integer	The unique BCM identifier for the report result.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ReportLoad

This class returns if a specific archived public report could be loaded and provides all its pertinent information. Depending on its type, not all of the parameters explained below are part of the class.

Parameter	Type	Description
ReportData	object[ <a href="#">ReportLoadReportData</a> ]	The array of type <code>ReportLoadReportData</code> , containing all information about the report, if the report is style-based. This parameter is not applicable to template-based reports.
Values	string	This parameter indicates the path to the report result file of the generated report. This parameter is not applicable to style-based reports.
groupId	integer	This number displays if the report is assigned to a BCM object, such as a device group or an operational rule, in which case this parameter returns the ID of the respective object, otherwise it returns 0. This parameter is not applicable to style-based reports.
ObjectID	integer	The unique BCM identifier of the report result.
CreateTime	string	The date and time the file was created on the originating device. This is the same information as that of the <b>Name</b> field part but in the default date and time format defined in the user preferences.
ReportType	string	Defines the type of the report, that is, if it is style or template based. This value of this parameter also defines which of the previously listed other parameters are applicable.

### ReportLoadReportData

This class returns the data of a specific style-based report.

Parameter	Type	Description
SubReportData	array[ <a href="#">ReportLoadSubReportData</a> ]	The array of type <code>ReportLoadSubReportData</code> , one for each subreport.
StyleFormat	string	The style of the report, that is, how many subreports the report has and their layout.
ReportTitle	string	The title of the report.
ReportDate	string	The date and time at which the report was generated.

Parameter	Type	Description
FontSize	integer	Defines the font size for this report. This may be a number between 8 and 18.
FontType	string	Defines the font type family for this report. This is also the font type used for the PDF generation.

### ReportLoadSubReportData

This class returns the subreport data.

Parameter	Type	Description
SubReportResult	integer	The error code of the subreport generation, which can be 0 for a successful generation or 1 if it failed.
FormatName	string	The type of the subreport, that is, if it is a table, a bar or pie or line chart.
Values	string	A string containing all subreport data.

## History

The history object type allows you to execute operations on specific log entries in the BCM database. These are generated by the console or by specific integration actions and are stored in a table. This group of actions is only used by external integrations such as BMC FootPrints Service Core and BMC Remedyforce.

[Logging history operations](#)

[Return Classes](#)

## Logging History Operations

The following operations are available on the logging history:

### GET

The following get operations are available:

Action	Description
<a href="#">/history/contexts</a>	Get logging history of an external integration.

### DELETE

The following delete operations are available:

Action	Description
<a href="#">/history/context/{contextIds}</a>	Delete specific entries of a external integration logging history.

## GET Operations

### /history/contexts

This operation recovers the logging data of all operations executed by a specific external integration in BCM, such as operational rules or remote control sessions.

### Syntax

GET /history/contexts?

language=OutputLanguage&fplogin=ExtIntAdminLogon&filterparameter=FPSC&offset=1&count=100

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.
filterparameter	query	no	string	The filter to apply on the context, to filter the log entries for the desired external integration.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

### Returns

Returns the class [ContextInfo](#).

### Possible Error Codes

The following errors can be returned:



Error Code	Description
0	Done.
16002	SQL execution error
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.
16223	Version not found
16224	The external interface version is older than the BMC Client Management interface version.
16225	The external interface version is newer than the BMC Client Management interface version.

## DELETE Operations

### /history/context/{contextIds}

This operation deletes specific entries of the logging history of a specific external integration.

### Syntax

```
DELETE /history/context/{contextIds}?fplogin=ExtIntAdminLogon
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
contextIds	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	array[int]	The unique BCM database identifier of the logging context(s) to delete. This value is provided by the GET <code>/history/contexts</code> operation.
externalLogin	string	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class [HistoryOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
16002	SQL execution error

Error Code	Description
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.
16223	Version not found
16224	The external interface version is older than the BMC Client Management interface version.
16225	The external interface version is newer than the BMC Client Management interface version.

## History Return Classes

The following classes can be returned by the different logging history actions:

Class	Description
<a href="#">ContextInfo</a>	The list of log entries matching the provided parameters.
<a href="#">HistoryOperationResult</a>	Log entry delete operation result.
<a href="#">RequestContext</a>	Data of a specific log entry.

### ContextInfo

This class returns the list of logging entries matching the demanded context for a specific external integration.

Parameter	Type	Description
ContextInfo	Array[ <a href="#">RequestContext</a> ]	This parameter returns one <a href="#">RequestContext</a> per log entry that matches the provided parameters.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### HistoryOperationResult

This class returns the result of the delete operation for specific log entries.

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### RequestContext

This class returns the available information about a log entry matching the specified parameters.

Parameter	Type	Description
ContextID	integer	The unique BCM database identifier of the logging context. This parameter is required by the <code>DELETE /history/context/{contextIds}</code> operation.
RequestContext	string	The request context, that is, the identifier of the external integration that was provided when the operation on BCM was executed.
Context	string	The type of the operation for which the log entry was created, for example, <i>operational rule</i> or <i>remote control</i> .
DeviceName	string	The name of the device on which the operation was executed.
OpRuleName	string	The name of the operational rule that was executed.
ErrorText	string	The error text of the operational rule, if one was returned.
Status	string	The execution status of the operational rule that was executed.
EndTime	string	The date and time at which the remote control session ended.
SessionDuration	integer	The total number of seconds that the remote control session was open.

## Cmdb

The cmdb object type allows you to extract from BMC Client Management and import these into external integration products such as BMC FootPrints Service Core views and BMC Remedyforce.

[Database operations](#)

[Return Classes](#)

## CMDB Database Operations

The following operations are available on the database:

### GET

The following get operations are available:

Action	Description
<a href="#">/cmdb/import</a>	Check if a data export is finished.

## POST

The following post operations are available:

Action	Description
<a href="#">/cmdb/import</a>	Launch BCM data export to a file.

## DELETE

The following delete operations are available:

Action	Description
<a href="#">/cmdb/datafile</a>	Delete external integration data import file.

## GET Operations

### [/cmdb/import](#)

This operation verifies if the export that is executed by another function has terminated. It loops until the final status of export terminated is returned.

### Syntax

```
GET /cmdb/import?datafile=path\import.cmdb&fplogin=ExtIntAdminLogin
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
datafile	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The name and full path of the database file into which the BCM data is to be exported.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class [CmdbDataResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
16002	SQL execution error
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.

## POST Operations

### /cmdb/import

This operation launches the data export of a specific BMC Client Management view, creates the output path and file into which the requested data is exported and makes it available for import into the database of an external integration, such as BMC FootPrints Service Core or BMC Remedyforce.

### Syntax

POST /cmdb/import?language=TargetLanguage&fplogin=ExtIntAdminLogin&cmdbData=path\file.cmdb

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
language	query	no	string	The language into which the internal BCM keywords are to be translated for the import, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.
cmdbData	body	no	CmdbObject	The data of a <code>CmdbObject</code> class that define the targets of the import and its desired data.

### Returns

Returns the class [CmdbOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
16002	SQL execution error
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.

## DELETE Operations

### /cmdb/datafile

This operation deletes a BMC Client Management database file after its data were imported into the external integration database.

### Syntax

```
DELETE /cmdb/datafile?fplogin=ExtIntAdminLogin&cmdbData=path\file.cmdb
```

### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
datafile	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The name and path of the database file to be deleted.
externalLogin	query	no	string	The login name of the external integration administrator who will execute the operation.

## Returns

Returns the class [CmdbOperationResult](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
16002	SQL execution error
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.

## Database Return Classes

The following classes can be returned by the different database actions:

Class	Description
<i>CmdbDataResult</i>	Return BCM data export result.
<i>CmdbExportResult</i>	This class returns all devices and device groups located directly at the root level of the main device groups node.
<i>CmdbObject</i>	Information required by an external integration database import.
<i>CmdbOperationResult</i>	Return the database delete operation result.
<i>DeviceType</i>	Provide device type list for data import.
<i>GroupID</i>	Provide device group list, if data import is limited to specific device groups.

### CmdbDataResult

This class returns the result of the BCM database export made available for other external integrations.

Parameter	Type	Description
Available	boolean	Indicates if the new BCM data record is now available for use in the external integration database.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### CmdbExportResult

This class returns the result and any necessary information of a database export.

Parameter	Type	Description
Url	string	The Url to download the database file.
FileName	string	The name and relative path of the database file.

Parameter	Type	Description
DeviceID	string	The list of device ids that were deleted since the last generated export file.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### CmdbObject

This class is used in the body of the POST /cmdb/import operation to provide the external integration with all required import information.

Parameter	Type	Description
lastImportTime	integer	The timestamp of the last import of the same data, to allow for the possibility of only importing changed data at the next import.
view	array[string]	The view of the target external integration into which the BCM data is to be imported.
types	array[DeviceType]	The types of devices if the data is not applicable to all devices.
groupIds	array[GroupID]	The ids of device groups, if the available data is only to be imported for one or more specific device groups.

### CmdbOperationResult

This class returns the result of a database operation.

Parameter	Type	Description
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### DeviceType

This class provides the list of device types for which the data is to be imported, if they are not applicable to all device types.

Parameter	Type	Description
deviceTypes	Array[string]	The types of devices for which the data is to be imported, if they are not applicable to all device types, such as <i>Laptop</i> , <i>Server</i> , <i>Workstation</i> , and so on.

## GroupID

This class provide the list of device groups, if the available data is only to be imported for one or more specific device groups.

Parameter	Type	Description
groupIds	array[int]	The unique BCM database identifier(s) of device groups, if the available data is only to be imported for one or more specific device groups. The ids of device groups are returned by the <code>GET /group/{groupId}/hierarchy</code> operation.

## Localization (I18N)

The i18n object type operations provide you the possibility to translate the cryptic error codes and other source code terms into legible text based on the localization of BCM and its available languages.

[I18N Operations](#)

[Return Classes](#)

## Localization (I18N) Operations

The following operations are available for the localization (i18n):

### GET

The following get operations are available:

Action	Description
<a href="#">/i18n/error/{errorId}</a>	Get Error code translation.

### POST

The following post operations are available:

Action	Description
<a href="#">/i18n/keywords</a>	Get keyword translation.

## GET Operations

### /i18n/error/{errorId}

This operation finds the textual translation of a specific error code in any of the languages supported by BMC Client Management.

### Syntax

GET /i18n/error/{errorId}?language=TargetLanguage

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
errorId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The code number of error code to translate, for example <i>16002</i> .

Name	Type	Mand.	Data Type	Description
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.

## Returns

Returns the class [ErrorCodeText](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

## Example

This example searches the text of error code number 16020 in Spanish.

### Request URL

```
https://localhost:1611/api/1/i18n/error/16020?language=Espanol
```

### Response Body

```
{ "language": "Espanol", "code": 16020, "text": "Nombre del tipo de grupo inválido." }
```

## POST Operations

### /i18n/keywords

This operation finds textual translation of keywords in any specified target language supported by BMC Client Management.

### Syntax

POST /i18n/keywords?language=TargetLanguage

The body must contain the definitions of the [KeywordsType](#) class.

```
{ "keywords": [ "string" ] }
```



#### Note

The keyword string can contain more than one value, the values must be separated with a colon (,).

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
keywords	body	no	[ <a href="#">KeywordsType</a> ]	This class contains the new data for the operational rule assignment update.
language	query	no	string	The language in which to display the results, which can be either English, American, Francais, Deutsch, Japanese, Espanol or Brasileiro. Select the desired value from the drop-down box.

## Returns

Returns the class [KeywordsTranslation](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

## Example

This example searches the translation of a number of top node names in German.

### Request URL

```
https://localhost:1611/api/1/i18n/keywords?language=Deutsch
```

whereby the body contains the following information:

```
{
  "keywords": [
    "_NODENAME_PROCESS_", "_NODENAME_PATCHGROUPRESULTS_", "_NODENAME_MODULECONFIG_", "_NODENAME_COMPLIANCERULERESULTS_"
  ]
}
```

### Response Body

```
{
  "language": "Deutsch",
  "keywords": {
    "_NODENAME_PROCESS_": "Prozeßverwaltung",
    "_NODENAME_PATCHGROUPRESULTS_": "Patchgruppenergebnisse",
    "_NODENAME_MODULECONFIG_": "Modulkonfiguration",

```

```

    "_NODENAME_COMPLIANCERULERESULTS_": "Compliance Regelergebnisse",
    "_NODENAME_CUSTINVOBJECTTYPES_": "Objekttypen des benutzerdefinierten Inventars"
  }
}

```

## Localization Return Classes

The following classes can be returned by the different localization actions:

Class	Description
<a href="#"><i>ErrorCodeText</i></a>	Return the localized error code text.
<a href="#"><i>KeywordsTranslation</i></a>	Return the list of translated keywords.
<a href="#"><i>KeywordsType</i></a>	Keyword list to translate.

### ErrorCodeText

This class returns the translation of the textual version of a specific error code in any of the languages supported by BMC Client Management.

Parameter	Type	Description
language	string	The target language in which the translation is requested.
code	integer	The error code for which the translation is required.
text	string	The translation of the error code in the target language.

### KeywordsTranslation

This class returns the lists of keywords and their translations in the requested language.

Parameter	Type	Description
language	string	The target language in which the translation is requested.
keywords	List[keywords]	The list of keywords with their translations in the requested target language.

### KeywordsType

This class lists the keywords that are to be translated to any of the languages supported by BMC Client Management.

Parameter	Type	Description
keywords	Array[string]	The list of keywords to translate. The keywords must be separated by a colon (,).



## Enums

The enums object type allows you to get all enumerations and their contents that are used.

[Enum operations](#)

[Return Classes](#)

## Enumerations

The following operations are available on enumerations:

### GET

The following get operations are available:

Action	Description
<a href="#">/enum/groups</a>	Get list of enumeration groups.
<a href="#">/enum/group</a>	The list of enumeration groups matching a search.

### GET Operations

#### [/enum/groups](#)

This operation list enumeration groups containing the query value provided.

### Syntax

GET /enum/groups?query={expression\_to\_contain}

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
query	query	no	string	The enumeration group to be searched. This search is based on contains operator.

### Returns

Returns the class [EnumGroups](#).

### Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
16002	SQL execution error
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.

## Example

This example searches enums that contain the value *Language*.

### Request URL

```
https://localhost:1611/api/1/enum/groups?query=Languages
```

### Response Body

```
{
  "groups": [
    {
      "name": "AvailableLanguages",
      "type": "String",
      "notes": ""
    },
    {
      "name": "PatchLanguages",
      "type": "String",
      "notes": ""
    }
  ]
}
```

Two enums match the search criteria, `AvailableLanguages` and `PatchLanguages` .

## /enum/group

This operation list enumerations for given enumeration group.

### Syntax

```
GET /enum/group?name={EnumGroupName}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
name	query	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The name of the enumeration group for which the enumerations are to be listed. The search is based on equality operator. This value is returned by the GET <code>/enum/groups</code> operation.

## Returns

Returns the class [Enumerations](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
16002	SQL execution error

Error Code	Description
16221	External Integration is not enabled for the specified user name. Ensure that the user name you specified matches the login specified in BMC Client Management under Global Settings > External Integration.

## Example

This example finds the values for the *AvailableLanguages* enum group.

### Request URL

```
https://localhost:1611/api/1/enum/group?name=AvailableLanguages
```

### Response Body

```
{
  "group": {
    "name": "AvailableLanguages",
    "members": [
      {
        "name": "_DB_LANGUAGE_ENGLISH_",
        "value": "English"
      },
      {
        "name": "_DB_LANGUAGE_FRENCH_",
        "value": "Francais"
      },
      {
        "name": "_DB_LANGUAGE_AMERICAN_",
        "value": "American"
      },
      {
        "name": "_DB_LANGUAGE_GERMAN_",
        "value": "Deutsch"
      },
      {
        "name": "_DB_LANGUAGE_JAPANESE_",
        "value": "Japanese"
      },
      {
        "name": "_DB_LANGUAGE_SPANISH_",
        "value": "Espanol"
      },
      {
        "name": "_DB_LANGUAGE_BRAZILIAN_",
        "value": "Brasileiro"
      }
    ]
  }
}
```

The return class lists all values that are contained in this group, in this case these are all the languages in which the console can be displayed.

## Enumeration Return Classes

The following classes can be returned by the different enumeration actions:

Class	Description
<a href="#">Enumerations</a>	The list of enumerations of a group.
<a href="#">EnumGroupDetails</a>	The details of an enumeration group.
<a href="#">EnumGroups</a>	Get list of enumeration groups.
<a href="#">EnumMembers</a>	The content of an enumeration.

## Enumerations

This class returns the list of enumeration groups containing a specific expression.

Parameter	Type	Description
Values	LIST[ <a href="#">EnumMembers</a> ]	Provides information about a specific child.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

## EnumGroupDetails

This class returns the details of a specific enumeration group.

Parameter	Type	Description
name	string	The name of the enumeration group.
type	string	The data type of the enumeration group.
notes	string	Any other information available for the enumeration group.

## EnumGroups

This class returns the list of enumeration groups containing a specific expression.

Parameter	Type	Description
Values	LIST[ <a href="#">EnumGroupDetails</a> ]	Provides information about a specific child.
ErrorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

## EnumMembers

This class returns the content of a specific enumeration.

Parameter	Type	Description
name	string	The name of the enumeration.
value	string	The value of the enumeration.

## Discovery

The asset discovery web services allow you can create scans and, using these scans on targets, discover assets.

[Discovery Operations](#)

[Discovery Return Classes](#)

## Discovery Operations

The following operations are available for the asset discovery:

### GET

The following get operations are available:

Action	Description
<a href="#">/discovery/target/lists/subfolders</a>	Get the list of folders at target root level.
<a href="#">/discovery/target/lists/folder/{folderId}/subfolders</a>	Get the list of subfolders of a target list folder.
<a href="#">/discovery/target/lists/folder/{folderId}</a>	Get the list of target lists of the specified folder.
<a href="#">/discovery/target/lists</a>	Get list of target lists at root folder.
<a href="#">/discovery/target/list/{targetListId}/members</a>	Get the members of the specified target list.
<a href="#">/discovery/scan/configurations/subfolders</a>	Get the list of folders at scan configuration root level.
<a href="#">/discovery/scan/configurations/folder/{folderId}/subfolders</a>	Get the list of subfolders of a scan configuration folder.
<a href="#">/discovery/scan/configurations/folder/{folderId}</a>	Get the list of scan configurations of the specified folder.
<a href="#">/discovery/scan/configurations</a>	Get list of scan configurations at root folder.
<a href="#">/discovery/scan/configuration/{configurationId}/protocols</a>	Get the protocols of the specified scan configuration.
<a href="#">/discovery/device/{deviceId}/scans/subfolders</a>	Get the list of subfolders at root level.
<a href="#">/discovery/scans/folder/{folderId}/subfolders</a>	Get the list of subfolders of a scan folder.
<a href="#">/discovery/device/{deviceId}/scans</a>	Get the list of scans at root level.
<a href="#">/discovery/scans/folder/{folderId}</a>	Get the list of scans under a folder.

Action	Description
<a href="#">/discovery/scan/{scanId}/configuration</a>	Get the configuration scan assigned to a scan.
<a href="#">/discovery/scan/{scanId}/target/lists</a>	Get the target lists assigned to a scan.
<a href="#">/discovery/scan/{scanId}/assignment</a>	Get the information related to a scanner assigned to a specific scan job.

## POST

The following post operation is available:

Action	Description
<a href="#">/discovery/scan/assignment/{assignmentId}</a>	Change the status of a scan job.

## PUT

The following put operations are available:

Action	Description
<a href="#">/discovery/target/lists/subfolder</a>	Create a target list folder under root.
<a href="#">/discovery/target/lists/folder/{folderId}/subfolder</a>	Create a target list subfolder in a specific folder.
<a href="#">/discovery/target/list</a>	Create a target list under root.
<a href="#">/discovery/target/list/folder/{folderId}</a>	Create a target list in a specific folder.
<a href="#">/discovery/target/list/{targetListId}/members</a>	Add members in a target list.
<a href="#">/discovery/scan/configurations/subfolder</a>	Create a scan configuration folder under root.
<a href="#">/discovery/scan/configurations/folder/{folderId}/subfolder</a>	Create a scan configuration subfolder in a specific folder.
<a href="#">/discovery/scan/configuration</a>	Create a scan configuration under root.
<a href="#">/discovery/scan/configuration/folder/{folderId}</a>	Create a scan configuration in a specific folder.
<a href="#">/discovery/scan/configuration/{configurationId}/protocol/{assignmentId}</a>	Modify a protocol in a scan configuration.
<a href="#">/discovery/scan/configuration/{configurationId}/protocols</a>	Modify all the protocols in a scan configuration.
<a href="#">/discovery/device/{deviceId}/scans/subfolder</a>	Create a scan folder under root.
<a href="#">/discovery/scans/folder/{folderId}/subfolder</a>	Create a scan subfolder in a specific folder.
<a href="#">/discovery/device/{deviceId}/scan</a>	Create a scan under root.

Action	Description
<a href="#">/discovery/scan/folder/{folderId}</a>	Create a scan under a folder.
<a href="#">/discovery/scan/{scanId}/configuration/{configurationId}</a>	Assign a scan configuration to a scan.
<a href="#">/discovery/scan/{scanId}/target/list/{targetListId}</a>	Assign a target list to a scan.
<a href="#">/discovery/scan/{scanId}/configuration/{configurationId}/target/list/{targetListId}</a>	Assign a configuration and/or target list to a scan.

## DELETE

The following delete operations are available:

Action	Description
<a href="#">/discovery/target/lists/folder/{folderId}</a>	Delete a target list folder.
<a href="#">/discovery/target/list/{targetListId}</a>	Delete a target list.
<a href="#">/discovery/target/list/{targetListId}/members/{assignmentId}</a>	Delete a target list member.
<a href="#">/discovery/scan/configurations/folder/{folderId}</a>	Delete a scan configuration folder.
<a href="#">/discovery/scan/configuration/{configurationId}</a>	Delete a scan configuration.
<a href="#">/discovery/scans/folder/{folderId}</a>	Delete a scan folder.
<a href="#">/discovery/scan/{scanId}</a>	Delete a scan.
<a href="#">/discovery/scan/{scanId}/configuration/{configurationId}</a>	Delete an assignment between a scan configuration and a scan.
<a href="#">/discovery/scan/{scanId}/target/list/{targetListId}</a>	Delete an assignment between a target list and a scan.

## Discovery GET Operations

### [/discovery/target/lists/subfolders](#)

This operation returns the list of target list folders located directly below the top node. When hierarchy parameter is set to true, this also lists targets at root level.

### Syntax

GET [discovery/target/lists/subfolders](#)?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowRules&offset=OffsetStart&count=EntriesPer

### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if target lists contained in the respective folder are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ObjectFolderArray](#).

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.

Error Code	Description
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds the following subfolders with further information and their hierarchy.

### Request URL

```
https://localhost:1611/api/1/discovery/target/lists/subfolders?
brief=false&sort=name&asc=true&hierarchy=false&offset=1&count=100
```

### Response Body

```
{
  "folders": [
    {
      "id": 271,
      "name": "France",
      "notes": "",
      "createDate": 1456929260,
      "updateDate": 1456979045
    }
  ],
  "total": 1
}
```

## **/discovery/target/lists/folder/{folderId}/subfolders**

This operation returns the list of subfolders of a specific target lists folder.

## Syntax

```
GET discovery/target/lists/folder/{folderId}/subfolders?
brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage
```

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target lists subfolder. This value is returned by either the <code>/target/lists/subfolders</code> or another <code>/target/lists/folder/{folderId}/subfolders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if target lists contained in the respective folder are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *ObjectFolderArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the subfolders of a target list folder with ID 271.

**Request URL**

```
https://localhost:1611/api/1/discovery/target/lists/folder/271/subfolders?
brief=false&sort=name&asc=true&count=100
```

**Response Body**

```
{
  "folders": [
    {
      "id": 274,
      "name": "Nice",
      "notes": "",
      "createDate": 1456979044,
      "updateDate": 1456979049
    },
    {
      "id": 273,
      "name": "Paris",
      "notes": "",
      "createDate": 1456979033,
      "updateDate": 1456979039
    }
  ],
  "total": 2
}
```

**/discovery/target/lists/folder/{folderId}**

This operation returns the list of target lists of a specific folder.

**Syntax**

GET discovery/target/lists/folder/{folderId}?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target lists folder. This value is returned by either the <code>/target/lists/subfolders</code> or another <code>/target/lists/folder/{folderId}/subfolders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the targetlist ( <code>name</code> ), the date at which the target list was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *TargetListArray*.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the target list available in the folder with ID 274.

### Request URL

```
https://localhost:1611/api/1/discovery/target/lists/folder/274?
brief=false&sort=name&asc=true&count=100
```

### Response Body

```
{
```

```

"targetlists": [
  {
    "id": 1002,
    "name": "Windows",
    "notes": "",
    "createDate": 1456981389,
    "updateDate": 1456981442
  }
],
"total": 1
}

```

## /discovery/target/lists

This operation returns the list of target lists directly located under the Target lists top node.

### Syntax

GET discovery/target/lists?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the target lists or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the targetlist ( <code>name</code> ), the date at which the target list was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

### Returns

Returns the class [TargetListArray](#).



## Comments

### Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example returns the target list available directly under the `Targets` node.

#### Request URL

```
https://localhost:1611/api/1/discovery/target/lists?brief=false&sort=name&asc=true&count=100
```

#### Response Body

```
{
  "targetlists": [
    {
      "id": 1000,
      "name": "Windows",
      "notes": "",
      "createDate": 1456880331,
      "updateDate": 1456880386
    }
  ],
  "total": 1
}
```

**/rollout/configuration/{configurationId}**

This operation gets the attributes of a rollout configuration.

**Syntax**

GET /rollout/configuration/{configurationId}

**Parameters**

The action has the following parameters:

Name	Path	Mand.	Data Type	allowMultiple	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	Indicates if multiple configurations are allowed or not, <i>true</i> if allowed, <i>false</i> if not allowed.	The unique BCM database identifier of the rollout configuration. This value is returned by either the GET /rollout/configurations or a GET /rollout/folder/{folderId}/configurations operation.

**Returns**

Returns the class *RolloutConfigurationAttrs*.

**Comments**

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.
16040	Bad object type.

## Example

This example returns the details of a configuration with ID 1012.

### Request URL

```
https://qacenter-srv2.bmc.com:7611/api/1/rollout/configuration/1012
```

### Response Body

```
{
  "rollout": {
    "configurationId": 1012,
    "name": "GL1_ROLLOUT_RELAY",
    "type": "_DB_ROLLOUTTYPE_INSTALL_",
    "os": "_OS_ROLLOUT_ALLWINDOWS_",
    "serviceName": "BMC Client Management Agent",
    "installDirectory": "D:\\BMC\\relay",
    "createDate": 1455670925,
    "installerName": "Relay_125.exe",
    "notes": "",
    "silentInstaller": false,
    "updateDate": 1455670934
  },
  "errorCode": 0
}
```

## /discovery/scan/configurations/subfolders

This operation returns the list of scan configuration folders located directly below the top node. When hierarchy parameter is set to true, this also lists scan configurations at root level.

### Syntax

GET /discovery/scan/configurations/subfolders?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowRules&offset=OffsetStart&count=EntriesPer

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
hierarchy	query	no	boolean	Select from this drop-down box if the scan configurations contained in the respective folder are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

### Returns

Returns the class [ObjectFolderArray](#). This class provides the number of children of a specific location and information on all these children.

### Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds the following subfolders with further information.

### Request URL

```
https://localhost:1611/api/1/discovery/scan/configurations/subfolders?
brief=false&sort=name&asc=true&hierarchy=false&count=100
```

### Response Body

```
{
  "folders": [
    {
      "id": 275,
      "name": "France",
      "notes": "",
      "createDate": 1456985758,
      "updateDate": 1456985790
    }
  ],
  "total": 1
}
```

## /discovery/scan/configurations/folder/{folderId}/subfolders

This operation returns the list of subfolders of a specific target lists folder.

### Syntax

```
GET /discovery/scan/configurations/folder/{folderId}/subfolders?
brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowRules&offset=OffsetStart&count=EntriesPerPage
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration subfolder. This value is returned by either the <code>/scan/configurations/subfolders</code> or another <code>/scan/configurations/folder/{folderId}/subfolders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if the scan configurations contained in the respective folder are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ObjectFolderArray](#). This class provides the number of children of a specific location and information on all these children.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the subfolders of a scan configuration folder with ID 275.

**Request URL**

```
https://localhost:1611/api/1/discovery/scan/configurations/folder/275/subfolders?
brief=false&sort=name&asc=true&hierarchy=false&count=100
```

**Response Body**

```
{
  "folders": [
    {
      "id": 274,
      "name": "Nice",
      "notes": "",
      "createDate": 1456979044,
      "updateDate": 1456979049
    },
    {
      "id": 273,
      "name": "Paris",
      "notes": "",
      "createDate": 1456979033,
      "updateDate": 1456979039
    }
  ],
  "total": 2
}
```

**/discovery/scan/configurations/folder/{folderId}**

This operation returns the list of scan configurations of a specific folder.

**Syntax**

```
GET discovery/discovery/scan/configurations/folder/{folderId}?
brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage
```

**Parameters**

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration folder. This value is returned by either the <code>/scan/configurations/subfolders</code> or another <code>/scan/configurations/folder/{folderId}/subfolders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the scan configuration or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the scan configuration ( <code>name</code> ), the date at which the scan configuration was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ScanConfigurationArray](#). This class returns the number of scan configurations found in the specific location and one scan configuration information array per configuration.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the scan configurations available in the folder with ID 275.

### Request URL

```
https://localhost:1611/api/1/discovery/scan/configurations/folder/275?
brief=false&sort=name&asc=true&count=100
```

### Response Body

```
{
  "configurations": [
    {
      "id": 1001,
      "name": "Windows",
      "notes": "",
      "createDate": 1456992683,
      "updateDate": 1456992697
    }
  ],
  "total": 1
}
```

## /discovery/scan/configurations

This operation returns the list of scan configurations directly located under the Scan configurations top node.

## Syntax

GET /discovery/scan/configurations?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the scan configurations or also additional available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the scan configuration ( <i>name</i> ), the date at which the scan configuration was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class [ScanConfigurationArray](#). This class returns the number of scan configurations found in the specific location and one scan configuration information array per configuration.

## Comments

### Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example returns the scan configurations available in directly under the `Scan Configurations` node.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/configurations?
brief=false&sort=name&asc=true&count=100
```

#### Response Body

```
{
  "configurations": [
    {
      "id": 1000,
      "name": "Windows 7",
      "notes": "",
      "createDate": 1456880330,
      "updateDate": 1456880354
    }
  ],
  "total": 1
}
```

## /discovery/scan/configuration/{configurationId}/protocols

This operation returns the protocols of a specific scan configuration.

### Syntax

GET /discovery/scan/configuration/{configurationId}/protocols

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration. This value is returned by either the <i>/scan/configurations</i> or a <i>/scan/configurations/folder/{folderId}/lists</i> operation.

### Returns

Returns the class [ScanConfigurationProtocolArray](#). This class returns the number of protocols found in the specific location and one protocol information array per protocol.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example returns the protocols for the scan configuration with ID 1000.

### Request URL

`https://localhost:1611/api/1/discovery/scan/configuration/1000/protocols`

### Response Body

```
{
  "Values": [
    {
      "AssignId": 1000,
      "IsActive": 1,
      "EnumValue": "SMB",
      "AdminLogin": "Administrateur:TnVtYXJhMDY=;",
      "Description": "Server Message Block",
      "IsAuth": 1,
      "HasEncPasswd": 1
    },
    {
      "AssignId": 1001,
      "IsActive": 1,
      "EnumValue": "SSH",
      "AdminLogin": "root:TnVtYXJhMDY=;",
      "Description": "Secure shell",
      "IsAuth": 1,
      "HasEncPasswd": 1
    }
  ],
}
```

```

{
  "AssignId": 1002,
  "IsActive": 1,
  "EnumValue": "SNMP",
  "AdminLogin": "community=cHVibGlj~user=~authalgo=~authpass=~privacyalgo=~privacypass=~;",
  "Description": "Simple Network Management Protocol",
  "IsAuth": 1,
  "HasEncPasswd": 1
},
{
  "AssignId": 1003,
  "IsActive": 1,
  "EnumValue": "VMWARE",
  "AdminLogin": "",
  "Description": "VM-Ware V-Sphere",
  "IsAuth": 1,
  "HasEncPasswd": 1
},
{
  "AssignId": 1004,
  "IsActive": 1,
  "EnumValue": "HYPERV",
  "AdminLogin": "",
  "Description": "Microsoft Hyper-V",
  "IsAuth": 1,
  "HasEncPasswd": 1
}
],
"ErrorCode": 0
}

```

### **/discovery/device/{deviceId}/scans/subfolders**

This operation returns the list of subfolders at the root level.

#### **Syntax**

GET /discovery/device/{deviceId}/scans/subfolders?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowRules&offset=OffsetStart&count=EntriesPerPage

#### **Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scanner.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also additional available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <i>name</i> ), the date at which the folder was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).



Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if the scan configurations contained in the respective folder are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *ObjectFolderArray*. This class provides the number of children of a specific location and information on all these children.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds the subfolders of a device with ID 1000.

### Request URL

```
https://localhost:1611/api/1/discovery/device/1000/scans/subfolders?
brief=false&sort=name&asc=true&hierarchy=false&count=100
```

### Response Body

```
{
  "folders": [
    {
      "id": 278,
      "name": "Windows 7",
      "notes": "",
      "createDate": 1456999584,
      "updateDate": 1456999594
    }
  ],
  "total": 1
}
```

### **/discovery/scans/folder/{folderId}/subfolders**

This operation returns the list of subfolders of a specific scan folder.

### Syntax

GET /discovery/scans/folder/{folderId}/subfolders?

brief=ShowDetails&sort=SortBy&asc=SortOrder&hierarchy=ShowRules&offset=OffsetStart&count=EntriesPerPage

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan subfolder. This value is returned by either the <code>/device/{deviceId}/scans/subfolders</code> or another <code>/scans/folder/{folderId}/subfolders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the subfolders or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the folder ( <code>name</code> ), the date at which the folder was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).

Name	Type	Mand.	Data Type	Description
hierarchy	query	no	boolean	Select from this drop-down box if the scans contained in the respective folder are listed in the return class, ( <code>true</code> ) for include the list, ( <code>false</code> ) for don't include.
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *ObjectFolderArray*. This class provides the number of children of a specific location and information on all these children.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds the subfolders of a scan folder with ID 278.

### Request URL

```
https://localhost:1611/api/1/discovery/scans/folder/278/subfolders?
brief=false&sort=name&asc=true&hierarchy=false&count=100
```

## Response Body

```
{
  "folders": [
    {
      "id": 279,
      "name": "Laptops",
      "notes": "",
      "createDate": 1457000397,
      "updateDate": 1457000404
    },
    {
      "id": 280,
      "name": "Workstations",
      "notes": "",
      "createDate": 1457000412,
      "updateDate": 1457000420
    }
  ],
  "total": 2
}
```

## /discovery/device/{deviceId}/scans

This operation returns the list of scans at the root level.

## Syntax

GET /discovery/device/{deviceId}/scans?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scanner.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the scans or also additional available information, <i>true</i> for IDs only, <i>false</i> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the scan ( <i>name</i> ), the date at which the scan was created ( <i>createDate</i> ) or the date at which it was last modified ( <i>updateDate</i> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <i>true</i> ) or descending ( <i>false</i> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.

Name	Type	Mand.	Data Type	Description
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *ObjectFolderArray*. This class provides the number of children of a specific location and information on all these children.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds the scans at root level of a device with ID 1000.

### Request URL

`https://localhost:1611/api/1/discovery/device/1000/scans?brief=false&sort=name&asc=true&count=100`

### Response Body

```
{
  "scans": [
    {
      "id": 1000,
      "name": "Laptop Scan",
      "notes": "",
      "createDate": 1456880330,
      "updateDate": 1456880351
    },
    {
```

```
{
  "id": 1001,
  "name": "Workstatioin Scan",
  "notes": "",
  "createDate": 1456999344,
  "updateDate": 1456999383
},
{
  "total": 2
}
```

### **/discovery/scans/folder/{folderId}**

This operation returns the list of scans under a folder.

### **Syntax**

GET /discovery/scans/folder/{folderId}?

brief=ShowDetails&sort=SortBy&asc=SortOrder&offset=OffsetStart&count=EntriesPerPage

### **Parameters**

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan subfolder. This value is returned by either the <code>/device/{deviceId}/scans/subfolders</code> or another <code>/scans/folder/{folderId}/subfolders</code> operation.
brief	query	no	boolean	Select from this drop-down box if the return class only returns the IDs of the scans or also additional available information, <code>true</code> for IDs only, <code>false</code> for IDs plus information such as name, creation date, and so on.
sort	query	no	integer	Select from this drop-down box the parameter according to which the returned values are to be sorted, this can either be the name of the scan ( <code>name</code> ), the date at which the scan was created ( <code>createDate</code> ) or the date at which it was last modified ( <code>updateDate</code> ).
asc	query	no	boolean	Select from this drop-down box in which order the sorting is applied, either ascending ( <code>true</code> ) or descending ( <code>false</code> ).
offset	query	no	integer	The offset from where on to start displaying the entries. Leave this field empty together with the <b>count</b> field to display all entries.

Name	Type	Mand.	Data Type	Description
count	query	no	integer	The number of entries to display at a time. Leave this field empty together with the <b>offset</b> field to display all entries.

## Returns

Returns the class *ObjectFolderArray*. This class provides the number of children of a specific location and information on all these children.

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

## Example

This example finds the scans available in a subfolder with ID 275.

### Request URL

```
https://localhost:1611/api/1/discovery/scans/folder/275?brief=false&sort=name&asc=true&count=100
```

### Response Body

```
{
  "scans": [
    {
      "id": 1001,
      "name": "Windows Discovery Scan",
      "notes": "",
      "createDate": 1456999344,
      "updateDate": 1456999383
    }
  ],
}
```

```
"total": 1  
}
```

### /discovery/scan/{scanId}/configuration

This operation returns the configuration scan assigned to a scan.

### Syntax

GET /discovery/scan/{scanId}/configuration

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the Get /device/{deviceId}/scans or a GET /scans/folder/{folderId} operation.

### Returns

Returns the class [ScanConfigurationAssign](#). This class provides the information on the scan configuration assigned to the scan.

Comments

Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

Example

This example returns the scan configurations assigned to a scan with ID 1000.

Request URL

https://localhost:1611/api/1/discovery/scan/1000/configuration

Response Body

```
{
  "configuration": [
    {
      "assignmentId": 1000,
      "Id": 1000,
      "name": "SmokeTests_Asset Discovery Configuration Scan"
    }
  ],
  "total": 1
}
```

/discovery/scan/{scanId}/target/lists

This operation returns the target lists assigned to a scan.

Syntax

GET /discovery/scan/{scanId}/target/lists

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the <code>Get /device/{deviceId}/scans</code> or a <code>GET /scans/folder/{folderId}</code> operation.

## Returns

Returns the class [TargetListScanAssignArray](#). This class provides the information on the target lists assigned to the scan.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example returns the target lists assigned to a scan with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/1000/target/lists
```

#### Response Body

```
{
  "targetlists": [
    {
      "assignmentId": 1000,
      "id": 1000,
      "name": "SmokeTests_Asset Discovery Target List"
    }
  ],
  "total": 1
}
```

### /discovery/scan/{scanId}/assignment

This operation returns the information related to a scanner assigned to a specific scan job.

### Syntax

```
GET /discovery/scan/{scanId}/assignment
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the <code>Get /device/{deviceId}/scans</code> or a <code>GET /scans/folder/{folderId}</code> operation.

## Returns

Returns the class [ScanAssignment](#). This class provides the information on the scanner assigned to the scan.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example returns the scanners assigned to a scan with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/1000/assignment
```

#### Response Body

```
{
  "scannerId": 1000,
  "scannerName": "MASTER-ORA",
  "assignmentId": 1000,
  "scheduleId": 1217,
  "status": "_STATUS_EXECUTEDOK_",
  "lastStatusUpdate": 1456881087,
  "errorCode": 0
}
```

## Discovery POST Operations

### /discovery/scan/assignment/{assignmentId}

This operation changes the status of a scan job.

#### Syntax

Post /discovery/scan/assignment/{assignmentId}?status=StatusToSet

The body must contain at least one of the status from the `StatusValues` class.

```
{ "status": { "type": "string", "allowableValues": { "valueType": "LIST",
"values": [ "_STATUS_ASSIGNPAUSED_", "_STATUS_UNASSIGNEDPAUSED_",
"_STATUS_ASSIGNWAITING_", "_STATUS_UPDATEWAITING_", "_STATUS_REASSIGNWAITING_",
"_STATUS_UNASSIGNWAITING_" ] } } }
```

#### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between a scan job and its scanner. This value is returned by <code>/scan/{scanId}/assignment</code> operation.

Name	Type	Mand.	Data Type	Description
status	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The list of statuses that can be set for the assignment, which can be any status from the <a href="#">StatusValues</a> class.

## Returns

Returns the class [OperationResult](#). This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example sets the assignment status as paused for a scan job with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/assignment/1000
```

#### Response Body

```
{
  "errorCode": 0
}
```

## Discovery PUT Operations

### /discovery/target/lists/subfolder

This operation creates a new target list folder directly below the Targets top node.

#### Syntax

```
Put /discovery/target/lists/subfolder
```

The body must contain the definitions of the `FolderAttrs` class.

```
{"name": "string", "notes": "string"}
```

#### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new folder to be created. For more information, see <a href="#">FolderAttrs</a> class.

## Returns

Returns the class [PutObjectFolderResult](#). This class returns the id of the newly created folder with error code.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new folder with name as France.

#### Request URL

```
https://localhost:1611/api/1/api/1/discovery/target/lists/subfolder
```

The body contains the following information of the `FolderAttrs` class.

```
{"name": "France", "notes": "France office target lists"}
```

#### Response Body

```
{
  "folder": {
    "id": 275
  }
}
```

### /discovery/target/lists/folder/{folderId}/subfolder

This operation creates a new target list subfolder in a specific target lists folder.

### Syntax

```
Put /discovery/target/lists/folder/{folderId}/subfolder
```

The body must contain the definitions of the `FolderAttrs` class.

```
{"name": "string", "notes": "string"}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier identifier of the target list subfolder. This value is returned by either the <code>/target/lists/subfolders</code> or another <code>/target/lists/folder/{folderId}/subfolders</code> operation.

Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new folder to be created. For more information, see <a href="#">FolderAttrs</a> class.

## Returns

Returns the class [ObjectFolderValues](#). This class provides the information available for a target lists folder.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new subfolder under a target list folder with ID 272.

#### Request URL

```
https://localhost:1611/api/1/api/1/discovery/target/lists/folder/272/subfolder
```

The body contains the following information of the `FolderAttrs` class for the new folder to be created.

```
{"name": "Lyon", "notes": "Lyon office target lists."}
```

#### Response Body

```
{
  "folder": {
    "id": 276
  }
}
```

### /discovery/target/list

This operation creates a new target list directly under the Targets top node.

### Syntax

```
Put /discovery/target/list
```

The body must contain the definitions of the `TargetListAttrs`, which in turn must contain the definition of the `MemberAttrs`.

```
{"name": "string", "targetMembers": [{ "members": "string", "excluded": "string"}]}
```

### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
TargetListAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new target list to be created. For more information, see <a href="#">TargetListAttrs</a> class.

## Returns

Returns the class [PutTargetListResult](#). This class returns the result of the create or update operation of the target list.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new target list named France with devices with IP range starting from 10.64.192.2 to 10.64.192.230 as members.

#### Request URL

```
https://localhost:1611/api/1/discovery/target/list
```

The body contains the following information of the `TargetListAttrs` class.

```
{"name": "France", "targetMembers": [{"members": "10.64.192.2-10.64.192.230"}]}
```

#### Response Body

```
{
  "targetlist": {
    "id": 1001
  },
  "members": [
    {
      "assignId": 1002
    }
  ]
}
```

### /discovery/target/list/folder/{folderId}

This operation creates a new target list in a specific folder.

### Syntax

```
Put /discovery/target/list/folder/{folderId}
```

The body must contain the definitions of the `TargetListAttrs`, which in turn must contain the definition of the `MemberAttrs`.

```
{"name": "string", "targetMembers": [{"members": "string", "excluded": "string"}]}
```

## Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target list subfolder. This value is returned by either the <i>/target/lists/subfolders</i> or another <i>/target/lists/folder/{folderId}/subfolders</i> operation.

Name	Type	Mand.	Data Type	Description
TargetListAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new target list to be created. For more information, see <a href="#">TargetListAttrs</a> class.

## Returns

Returns the class [PutTargetListResult](#). This class returns the result of the create or update operation of the target list.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new target list named Lyon with devices with IP range starting from 10.64.192.2 to 10.64.192.230 as members under a folder with ID 272.

#### Request URL

```
https://localhost:1611/api/1/discovery/target/list/folder/272
```

The body contains the following information of the `TargetListAttrs` class.

```
{"name": "Lyon", "targetMembers": [{"members": "10.64.192.2-10.64.192.230"}]}
```

#### Response Body

```
{
  "targetlist": {
    "id": 1002
  },
  "members": [
    {
      "assignId": 1003
    }
  ]
}
```

### /discovery/target/list/{targetListId}/members

This operation adds members in a specific target list.

### Syntax

```
Put /discovery/target/list/{targetListId}/members
```

The body must contain the definitions of the `MembersAttrsArray`, which in turn must contain the definition of the `MemberAttrs`.

```
{"targetMembers": [{"members": "string", "excluded": "string"}]}
```

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
targetListId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target list. This value is returned by either the <code>GET /target/lists</code> or a <code>GET /target/folder/{folderId}/lists</code> operation.

Name	Type	Mand.	Data Type	Description
MembersAttrsArray	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new memberse to be added to the target list. For more informatioin, see <a href="#">MembersAttrsArray</a> class.

## Returns

Returns the class [AssignValuesResult](#). This class returns the assign id of the link between the target list and the member.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example adds a new device member with IP 10.64.192.231 in the target list with ID 1002.

#### Request URL

```
https://localhost:1611/api/1/discovery/target/list/1002/members
```

The body contains the following information of the `MembersAttrsArray` class.

```
{"targetMembers": [{"members": "10.64.192.231"}]}
```

#### Response Body

```
{
  "members": [
    {
      "assignId": 1004
    }
  ]
}
```

### /discovery/scan/configurations/subfolder

This operation creates a new scan configuration folder directly below the Scan configurations top node.

### Syntax

```
Put /discovery/scan/configurations/subfolder
```

The body must contain the definitions of the `FolderAttrs` class.

```
{"name": "string", "notes": "string"}
```

### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new folder to be created. For more information, see <a href="#">FolderAttrs</a> class.

## Returns

Returns the class [PutObjectFolderResult](#). This class returns the id of the newly created folder with error code.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new folder with name as US.

#### Request URL

```
https://localhost:1611/api/1/api/1/discovery/scan/configurations/subfolder
```

The body contains the following information of the `FolderAttrs` class.

```
{"name": "US", "notes": "US office target lists"}
```

#### Response Body

```
{
  "folder": {
    "id": 277
  }
}
```

### /discovery/scan/configurations/folder/{folderId}/subfolder

This operation creates a new scan configuration subfolder in a specific scan configurations folder.

### Syntax

```
Put /discovery/scan/configurations/folder/{folderId}/subfolder
```

The body must contain the definitions of the `FolderAttrs` class.

```
{"name": "string", "notes": "string"}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration subfolder. This value is returned by either the <code>/scan/configurations/subfolders</code> or another <code>/scan/configurations/folder/{folderId}/subfolders</code> operation.

Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new folder to be created. For more information, see <a href="#">FolderAttrs</a> class.

## Returns

Returns the class [ObjectFolderValues](#). This class provides the information available for a scan configurations folder.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new subfolder under a scan configuration folder with ID 277.

#### Request URL

```
https://localhost:1611/api/1/api/1/discovery/scan/configurations/folder/277/subfolder
```

The body contains the following information of the `FolderAttrs` class for the new folder to be created.

```
{"name": "Austin", "notes": " Scan configurations for the Austin office."}
```

#### Response Body

```
{
  "folder": {
    "id": 281
  }
}
```

### /discovery/scan/configuration

This operation creates a new scan configuration directly under the scan configurations top node.

### Syntax

```
Put /discovery/scan/configuration
```

The body must contain the definitions of the `ScanConfigurationAttrs` class.

```
{"name": "string", "notes": "additionalInformation"}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
ScanConfigurationAttrBody	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new scan configuration to be created. For more information, see <a href="#">ScanConfigurationAttrs</a> class.

## Returns

Returns the class [PutScanConfigurationResult](#). This class returns the result of the create or update operation of the scan configuration.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new scan configuration named France.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/configuration
```

The body contains the following information of the `ScanConfigurationAttrs` class.

```
{"name": "France", "notes": "Scan configurations for devices in France."}
```

#### Response Body

```
{
  "configuration": {
    "id": 1001
  }
}
```

### /discovery/scan/configuration/folder/{folderId}

This operation creates a new scan configuration in a specific folder.

### Syntax

```
Put /discovery/scan/configuration/folder/{folderId}
```

The body must contain the definitions of the `ScanConfigurationAttrs` class.

```
{"name": "string", "notes": "additionalInformation"}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration subfolder. This value is returned by either the <code>/scan/configurations/subfolders</code> or another <code>/scan/configurations/folder/{folderId}/subfolders</code> operation.



Name	Type	Mand.	Data Type	Description
ScanConfigurationAttrBody	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new scan configuration to be created. For more information, see <a href="#">ScanConfigurationAttrs</a> class.

## Returns

Returns the class [PutScanConfigurationResult](#). This class returns the result of the create or update operation of the scan configuration.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new scan configuration named Paris under the folder with ID 277.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/configuration/folder/{folderId}
```

The body contains the following information of the `ScanConfigurationAttrs` class.

```
{"name": "Paris", "notes": "Scan configurations for devices in Paris office."}
```

#### Response Body

```
{
  "configuration": {
    "id": 1006
  }
}
```

### /discovery/scan/configuration/{configurationId}/protocol/{assignmentId}

This operation modifies protocols in a specific scan configuration.

### Syntax

```
Put /discovery/scan/configuration/{configurationId}/protocol/{assignmentId}
```

The body must contain the modified definitions of the `ProtocolAttrs` class, which needs to be set.

```
{"activated": "int", "credentials": [{"login": "string", "password": "string"}]}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration. This value is returned by either the GET /scan/configurations or a GET /scan/configurations/folder/{folderId} operation.

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the link between the scan configuration and the protocol to modify. This value is returned by the <code>GET /scan/configuration/{configurationId}/protocols</code> operation.

Name	Type	Mand.	Data Type	Description
ProtocolAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The modified attributes for protocol to be set. For more information, see <a href="#">ProtocolAttrs</a> class.

## Returns

Returns the class [ProtocolAssignValuesResult](#). This class returns the assign id of the link between the scan configuration and the protocol.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.

Error Code	Description
1	Operation failed.
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example modifies the protocols assigned with ID 1000 to configuration with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/configuration/1000/protocol/1000
```

The body contains the following information of the `ProtocolAttrs` class.

```
{"activated": "0", "credentials": [{"login": "administrator", "password": "administrator"}]}
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/scan/configuration/{configurationId}/protocols

This operation modifies all the protocols in a specific scan configuration.

### Syntax

```
Put /discovery/scan/configuration/{configurationId}/protocols
```

The body must contain the modified definitions of the `AllProtocolAttrs` class.

```
{"protocol": [{"activated": "int", "credentials": [{"login": "string", "password": "string"}]}]}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration. This value is returned by either the GET /scan/configurations or a GET /scan/configurations/folder/{folderId} operation.

Name	Type	Mand.	Data Type	Description
AllProtocolAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The one array with modified attributes for each protocol to be modified. For more information, see <a href="#">AllProtocolAttrs</a> class.

## Returns

Returns the class [OperationResult](#). This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.



Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example modifies the protocols of the configuration with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/configuration/1000/protocols
```

The body contains the following information of the `AllProtocolAttrs` class.

```
{"SMB": [{"activated": "0", "credentials": [{"login": "SMBadministrator", "password": "administrator"}]}]}
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/device/{deviceId}/scans/subfolder

This operation creates a new scan folder directly below the Scanner top node.

### Syntax

```
Put /discovery/device/{deviceId}/scans/subfolder
```

The body must contain the definitions of the `FolderAttrs` class.

```
{"name": "string", "notes": "string"}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scanner.

Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new folder to be created. For more information, see <a href="#">FolderAttrs</a> class.

## Returns

Returns the class [PutObjectFolderResult](#). This class returns the result of a scan folder creation operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new folder with name as Windows under the scanner with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/api/1/discovery/device/1000/scans/subfolder
```

The body contains the following information of the `FolderAttrs` class.

```
{"name": "Windows", "notes": "Scan all Windows devices."}
```

#### Response Body

```
{
  "ObjectId": 278
}
```

### /discovery/scans/folder/{folderId}/subfolder

This operation creates a new scan subfolder in a specific scan folder.

### Syntax

```
Put /discovery/scans/folder/{folderId}/subfolder
```

The body must contain the definitions of the `FolderAttrs` class.

```
{"name": "string", "notes": "string"}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scanner.

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan subfolder. This value is returned by either the <i>/device/{deviceId}/scans/subfolders</i> or another <i>/scans/folder/{folderId}/subfolders</i> operation.

Name	Type	Mand.	Data Type	Description
FolderAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new folder to be created. For more information, see <a href="#">FolderAttrs</a> class.

## Returns

Returns the class [ObjectFolderValues](#). This class provides the information available for a scan folder.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new folder with name as Windows 7 under a folder with ID 278 under the scanner with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/api/1/discovery/scans/folder/278/subfolder
```

The body contains the following information of the `FolderAttrs` class.

```
{"name": "Windows 7", "notes": "Scan all Windows 7 devices."}
```

#### Response Body

```
{
  "ObjectId": 278
}
```

### /discovery/device/{deviceId}/scan

This operation creates a new scan directly below the Scanner top node.

### Syntax

```
Put /discovery/device/{deviceId}/scan
```

The body must contain the definitions of the `ScanAttrs` class.

```
{"name": "string", "notes": "additionalInformation"}
```

### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
deviceId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scanner.

Name	Type	Mand.	Data Type	Description
ScanAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new scan to be created. For more information, see <a href="#">ScanAttrs</a> class.

## Returns

Returns the class [ScanResult](#). This class returns the result of a scan creation operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new scan named Windows-Master on the device with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/discovery/device/1000/scan
```

The body contains the following information of the `ScanAttrs` class.

```
{"name": "Windows-Master", "notes": "Scans all Windows devices."}
```

#### Response Body

```
{
  "scan": {
    "id": 1001,
    "scheduleId": 1244
  }
}
```

### /discovery/scan/folder/{folderId}

This operation creates a new scan under a folder.

### Syntax

```
Put /discovery/scan/folder/{folderId}
```

The body must contain the definitions of the `ScanAttrs` class.

```
{"name": "string", "notes": "additionalInformation"}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan subfolder. This value is returned by either the <code>/device/{deviceId}/scans/subfolders</code> or <code>/scans/folder/{folderId}/subfolders</code> operation.

Name	Type	Mand.	Data Type	Description
ScanAttrs	body	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	string	The data for the new scan to be created. For more information, see <a href="#">ScanAttrs</a> class.

## Returns

Returns the class [PutObjectFolderResult](#). This class returns the result of a scan creation operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example creates a new scan named Windows-Servers under a folder with ID 278.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/folder/278
```

The body contains the following information of the `ScanAttrs` class.

```
{"name": "Windows-Server", "notes": "Scans all Windows servers."}
```

#### Response Body

```
{
  "scan": {
    "id": 1002
  }
}
```

### **/discovery/scan/{scanId}/configuration/{configurationId}**

This operation assigns a specific scan configuration to a specific scan.

### Syntax

```
Put /discovery/scan/{scanId}/configuration/{configurationId}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the GET /device/{deviceId}/scans or a GET /scans/folder/{folderId} operation.

Name	Type	Mand.	Data Type	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration. This value is returned by either the GET /scan/configurations or a GET /scan/configurations/folder/{folderId} operation.

## Returns

Returns the class *OperationResult*. This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.



Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example assigns a scan configuration with ID 1000 to a scan with ID 1002.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/1002/configuration/1000
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/scan/{scanId}/target/list/{targetListId}

This operation assigns a specific target list to a specific scan.

### Syntax

```
Put /discovery/scan/{scanId}/target/list/{targetListId}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the GET /device/{deviceId}/scans or a GET /scans/folder/{folderId} operation.

Name	Type	Mand.	Data Type	Description
targetListId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target list. This value is returned by either the GET /target/lists or a GET /target/folder/{folderId}/lists operation.

## Returns

Returns the class *OperationResult*. This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

**Example**

This example assigns a target list with ID 1001 to a scan with ID 1002.

**Request URL**

`https://localhost:1611/api/1/discovery/scan/1002/target/list/1001`

**Response Body**

```
{
  "errorCode": 0
}
```

**/discovery/scan/{scanId}/configuration/{configurationId}/target/list/{targetListId}**

This operation assigns a specific scan configuration and/or a specific target list to a specific scan.

**Syntax**

`Put /discovery/scan/{scanId}/configuration/{configurationId}/target/list/{targetListId}`

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the GET /device/{deviceId}/scans or a GET /scans/folder/{folderId} operation.

Name	Type	Mand.	Data Type	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration. This value is returned by either the GET /scan/configurations or a GET /scan/configurations/folder/{folderId} operation.

Name	Type	Mand.	Data Type	Description
targetListId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target list. This value is returned by either the GET /target/lists or a GET /target/folder/{folderId}/lists operation.

## Returns

Returns the class *OperationResult*. This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example assigns a target list with ID 1001 and scan configuration with ID 1000 to a scan with ID 1002.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/1002/configuration/1000/target/list/1001
```

#### Response Body

```
{
  "errorCode": 0
}
```

### Discovery DELETE Operations

#### /discovery/target/lists/folder/{folderId}

This operation deletes a specific target list folder.

#### Syntax

```
Delete /discovery/target/lists/folder/{folderId}
```

#### Parameters

The action has the following parameters:



Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target lists subfolder. This value is returned by either the <i>/target/lists/subfolders</i> or a <i>/target/lists/folder/{folderId}/subfolders</i> operation.

## Returns

Returns the class [OperationResult](#). This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

**Example**

This example deletes the target list folder with ID 271.

**Request URL**

`https://localhost:1611/api/1/discovery/target/lists/folder/271`

**Response Body**

```
{
  "errorCode": 0
}
```

**/discovery/target/list/{targetListId}**

This operation deletes a specific target list.

**Syntax**

Delete /discovery/target/list/{targetListId}

**Parameters**

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
targetListId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target list. This value is returned by either the GET /target/lists or a GET /target/folder/{folderId}/lists operation.

## Returns

Returns the class [OperationResult](#). This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example deletes the target list with ID 1002.

#### Request URL

```
https://localhost:1611/api/1/discovery/target/list/1002
```

#### Response Body

```
{
  "errorCode": 0
}
```

### **/discovery/target/list/{targetListId}/members/{assignmentId}**

This operation deletes a specific target list member.

### Syntax

Delete /discovery/target/list/{targetListId}/members/{assignmentId}

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
targetListId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target list. This value is returned by either the GET /target/lists or a GET /target/folder/{folderId}/lists operation.

Name	Type	Mand.	Data Type	Description
assignmentId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the assignment between the target list and the target list member. This value is returned by the GET <code>/target/list/{targetListId}/members</code> operation.

## Returns

Returns the class [OperationResult](#). This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example deletes the target list member with assignment ID 1005 from the target list with ID 1000.

#### Request URL

```
https://localhost:1611/api/1/discovery/target/list/1000/members/1005
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/scan/configurations/folder/{folderId}

This operation deletes a specific scan configuration folder.

### Syntax

Delete /discovery/scan/configurations/folder/{folderId}

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration subfolder. This value is returned by either the <code>Get /scan/configurations/subfolders</code> or a <code>Get /scan/configurations/folder/{folderId}/subfolders</code> operation.

## Returns

Returns the class [OperationResult](#). This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.



Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example deletes the scan configuration folder with ID 280.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/configurations/folder/280
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/scan/configuration/{configurationId}

This operation deletes a specific scan configuration.

### Syntax

Delete /discovery/scan/configuration/{configurationId}

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration. This value is returned by either the GET <i>/scan/configurations</i> or a GET <i>/scan/configurations/folder/{folderId}</i> operation.

Returns

Returns the class *OperationResult*. This class returns the result (error code) of the operation.

Comments

Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example deletes the scan configuration with ID 1006.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/configuration/1006
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/scans/folder/{folderId}

This operation deletes a specific scan folder.

### Syntax

```
Delete /discovery/scans/folder/{folderId}
```

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
folderId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan subfolder. This value is returned by either the <code>Get /device/{deviceId}/scans/subfolders</code> or a <code>Get /scans/folder/{folderId}/subfolders</code> operation.

Returns

Returns the class *OperationResult*. This class returns the result (error code) of the operation.

Comments

Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example deletes the scan folder with ID 281.

#### Request URL

```
https://localhost:1611/api/1/discovery/scans/folder/281
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/scan/{scanId}

This operation deletes a specific scan.

### Syntax

Delete /discovery/scan/{scanId}

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the GET /device/{deviceId}/scans or a GET /scans/folder/{folderId} operation.

Returns

Returns the class *OperationResult*. This class returns the result (error code) of the operation.

Comments

Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example deletes the scan with ID 1002.

#### Request URL

```
https://localhost:1611/api/1/discovery/scan/1002
```

#### Response Body

```
{
  "errorCode": 0
}
```

### /discovery/scan/{scanId}/configuration/{configurationId}

This operation deletes a specific scan.

### Syntax

Delete /discovery/scan/{scanId}/configuration/{configurationId}

### Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the GET /device/{deviceId}/scans or a GET /scans/folder/{folderId} operation.



Name	Type	Mand.	Data Type	Description
configurationId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan configuration. This value is returned by either the GET /scan/configurations or a GET /scan/configurations/folder/{folderId} operation.

## Returns

Returns the class [OperationResult](#). This class returns the result (error code) of the operation.

## Comments

## Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

Example

This example deletes the assignment of scan configuration with ID 1000 to scan with ID 1001.

Request URL

https://localhost:1611/api/1/discovery/scan/1001/configuration/1002

Response Body

```
{
  "errorCode": 0
}
```

/discovery/scan/{scanId}/target/list/{targetListId}

This operation deletes an assignment between a specific target list and a specific scan.

Syntax

Delete /discovery/scan/{scanId}/target/list/{targetListId}

Parameters

The action has the following parameters:

Name	Type	Mand.	Data Type	Description
scanId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the scan. This value is returned by either the GET /device/{deviceId}/scans or a GET /scans/folder/{folderId} operation.

Name	Type	Mand.	Data Type	Description
targetListId	path	The URL to the web page or internal share that is opened by the operational rule, for example, <i>http://www.google.com</i> or <i>//myshare.myserver.com/share1/share01</i> . This parameter is only applicable for Quick Link rules.	integer	The unique BCM database identifier of the target list. This value is returned by either the GET /target/lists or a GET /target/folder/{folderId}/lists operation.

Returns

Returns the class *OperationResult*. This class returns the result (error code) of the operation.

Comments

Possible Error Codes

The following errors can be returned:

Error Code	Description
0	Done.
1	Operation failed.

Error Code	Description
10	The requested object could not be found.
12	Access denied. Please verify your access rights.
13	Incorrect parameter. Please verify the supplied parameters.
16002	SQL execution error
16003	No handler found for the required operation.
16010	Login failed.

### Example

This example deletes the assignment of target list with ID 1002 to scan with ID 1001.

#### Request URL

`https://localhost:1611/api/1/discovery/scan/1001/target/list/1002`

#### Response Body

```
{
  "errorCode": 0
}
```

## Discovery Return Classes

The following classes can be returned by the different discovery actions:

Class	Description
<i>AllProtocolAttrs</i>	Provides information about all the protocols available for a scan configuration.
<i>AssignValuesResult</i>	Returns the unique BCM database identifier of assignment that links the target list and the member.
<i>CredentialAttrs</i>	Provides the credentials for a protocol of a scan configuration.
<i>FolderAttrs</i>	Provides information required to create or update an operational rules folder.
<i>FolderValuesResult</i>	Returns the result of the create or update operation of an object folder.
<i>MembersAttrsArray</i>	Provides an array with the attribute parameters and their values for the members that are to be included in or excluded from the target list.

Class	Description
<i>MemberAttrs</i>	Provides the attribute parameters and their values that be specified for the members to be included in or excluded from the target list.
<i>ObjectFolderArray</i>	Returns the number of children of a specific location and information about all these children.
<i>ObjectFolderValuesArray</i>	Returns the result of the create or update operation of an object folder.
<i>OjectFolderValues</i>	Returns returns the information available for an object folder.
<i>OperationResult</i>	Returns the result (error code) of the operation.
<i>ProtocolAssignValuesResult</i>	Returns the unique id of the protocol assignment value that links the scan configuration and the protocol.
<i>ProtocolAttrs</i>	Provides the protocol status and credentials of a scan configuration.
<i>PutObjectFolderResult</i>	Provides the result of the create or update operation of an object folder.
<i>PutScanConfigurationResult</i>	Provides the result of the create or update operation of a scan configuration.
<i>PutTargetListResult</i>	Provides the result of the create or update operation of a target list.
<i>ScanAssignment</i>	Provides the information on the scanners assigned to the scan.
<i>ScanAttrs</i>	Provides the information that is necessary to create a new asset discovery scan.
<i>ScanConfigurationArray</i>	Returns the number of scan configurations found in the specific location and one scan configuration information array per scan configuration.
<i>ScanConfigurationAssign</i>	Returns the information on the scan configuration assigned to the scan.
<i>ScanConfigurationAttrs</i>	Provides the information that is necessary to create a new scan configuration.
<i>ScanConfigurationListValues</i>	Provides one array for each scan configuration.
<i>ScanConfigurationProtocolArray</i>	Returns the protocols of the specified scan configuration.
<i>ScanConfigurationProtocolValues</i>	Returns one array for each scan configuration protocol.
<i>ScanConfigurationProtocolValuesArray</i>	Returns the number of scan configuration protocols found for a scan configuration and one array for each protocol.

Class	Description
<a href="#"><i>ScanConfigurationValuesArray</i></a>	Returns the information that is available for a scan configuration.
<a href="#"><i>ScanConfigurationValuesResult</i></a>	Returns the result of the create or update operation of the scan configuration.
<a href="#"><i>ScanResult</i></a>	Returns the result of the scan.
<a href="#"><i>SNMPCredentialAttrs</i></a>	Provides the information that is necessary to access the SNMP protocol of a scan configuration.
<a href="#"><i>SNMPProtocolAttrs</i></a>	Provides the SNMP protocol status and credentials of a scan configuration.
<a href="#"><i>StatusValues</i></a>	Provides information about different assignment statuses.
<a href="#"><i>TargetAssignIds</i></a>	Provides the unique id of the assignment link between the target list and members.
<a href="#"><i>TargetListArray</i></a>	Returns the number of target lists found in the specific location and one target list information array per target list.
<a href="#"><i>TargetListAttrs</i></a>	Provides the information that is necessary to create a new target list.
<a href="#"><i>TargetListMemberArray</i></a>	Returns the number of target list members found in the specific location and one member information array per list.
<a href="#"><i>TargetListMemberValues</i></a>	Returns one array for each member of the target list.
<a href="#"><i>TargetListMemberValuesArray</i></a>	Returns the information that is available for a target list member.
<a href="#"><i>TargetListScanAssignArray</i></a>	Returns the information on the target lists assigned to the scan.
<a href="#"><i>TargetListScanAssignValues</i></a>	Returns one array for each scan assigned to the target list.
<a href="#"><i>TargetListScanAssignValuesArray</i></a>	Returns the information available for each scan assigned to a target list.
<a href="#"><i>TargetListValues</i></a>	Returns one array for each target list.
<a href="#"><i>TargetListValuesArray</i></a>	Returns the information that is available for a target list.
<a href="#"><i>TargetListValuesResult</i></a>	Returns the result of the create or update operation of the target list.

### AllProtocolAttrs

This class provides the information about all the protocols available for a scan configuration. It is required by the PUT / discovery/scan/configuration/{configurationId}/protocols operation.

Parameter	Type	Description
SMB	array[ <a href="#">ProtocolAttrs</a> ]	SMB protocol attributes.
SSH	array[ <a href="#">ProtocolAttrs</a> ]	SSH protocol attributes.
SNMP	array[ <a href="#">ProtocolAttrs</a> ]	SNMP protocol attributes.
VMWARE	array[ <a href="#">ProtocolAttrs</a> ]	VMWARE protocol attributes.
HYPERV	array[ <a href="#">ProtocolAttrs</a> ]	HYPERV protocol attributes.

### AssignValuesResult

This class returns the unique id of the assignment value that links the target list and the member. It is used by the `PUT /discovery/target/list/{targetListId}/members` operation.

Parameter	Type	Description
Id	integer	The id of the assign value linking the target list and the member.

### CredentialAttrs

This class provides the credentials for a protocol of a scan configuration. It is required by the `PUT /discovery/scan/configuration/{configurationId}/protocol/{assignmentId}` and `PUT /discovery/scan/configuration/{configurationId}/protocols` operations.

Parameter	Type	Description
login	string	The login for the protocol.
password	string	The password for the specified login for the protocol.

### FolderAttrs

This class provides the information required to create or update an object folder.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.

### FolderValuesResult

This class returns the result of the create or update operation of an object folder.



Parameter	Type	Description
Id	integer	The unique BCM database identifier of the object folder.

### MembersAttrsArray

This class provides the assign IDs between the target list and each member one information array per member.

Parameter	Type	Description
assignmentId	Array[ <a href="#">TargetAssignIds</a> ]	One array for each member.

### MemberAttrs

This class provides the attribute parameters and their values that be specified for the members to be included in or excluded from the target list. It is required by the PUT /discovery/target/list/folder/{folderId} and PUT /discovery/target/list operations.

Parameter	Type	Description
members	string	Enter the name of the device to be added to the list. The name may be entered either as its short or long network name, for example, <i>scotty</i> or <i>scotty.enterprise.com</i> or as its IP address, for example, <i>159.124.5.10</i> or <i>2001:db8:85a3::8a2e:370:7334</i> , or as a comma separated list of names and/or ranges, for example, <i>scotty,192.168.4.45-192.168.4.47,2001:db8:85a3::8a2e:370:7334</i> which includes machines <i>scotty.enterprise.com</i> , <i>2001:db8:85a3::8a2e:370:7334</i> , <i>192.168.4.45</i> , <i>192.168.4.46</i> and <i>192.168.4.47</i> . A range may also be entered as CIDR notation in the form of <i>192.9.205.22/18</i> or <i>2001:db8:85a3::8a2e:370:152/896</i> . Be aware that the address <i>x.x.x.255</i> , being a broadcast IP address cannot be scanned and therefore cannot be included in any target range.
excluded	string	This free text field may contain the devices which are to be excluded from the range to scan defined in members field. The same naming rules are applied as specified for members field.

### ObjectFolderArray

This class returns the number of children of a specific location and information about all these children. It is returned by the GET /discovery/target/lists/subfolders and GET /discovery/target/lists/folder/{folderId}/subfolders operations.

Parameter	Type	Description
total	integer	The total number of object folders found in the specified location.
values	Array[ <a href="#">ObjectFolderValues</a> ]	The available information for an object folder.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ObjectFolderValuesArray

This class provides the information available for an object folder.

Parameter	Type	Description
Id	integer	The unique BCM database ID of the object folder.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

### ObjectFolderValues

This class returns the information available for an object folder.

Parameter	Type	Description
folder	array[ <a href="#">ObjectFolderValuesArray</a> ]	One array for each object folder that is found in this location.

### OperationResult

This class returns the result of an operation.

Parameter	Type	Description
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ProtocolAssignValuesResult

This class returns the unique id of the protocol assignment value that links the scan configuration and the protocol. It is used by the `PUT /discovery/scan/configuration/{configurationId}/protocol/{assignmentId}` operation.

Parameter	Type	Description
Id	integer	The id of the assign value linking the scan configuration and the protocol.

### ProtocolAttrs

This class provides the protocol status and credentials of a scan configuration. It is required by the `PUT /discovery/scan/configuration/{configurationId}/protocol/{assignmentId}` and `PUT /discovery/scan/configuration/{configurationId}/protocols` operations.

Parameter	Type	Description
activated	integer	Indicates if the protocol is activated.
credentials	array[ <a href="#">CredentialAttrs</a> ]	The login and password for the activated protocol.

### PutObjectFolderResult

This class provides the result of the create or update operation of an object folder. It is returned by the `PUT /discovery/target/lists/subfolder`, `PUT /discovery/scan/configurations/subfolder`, `PUT /discovery/device/{deviceId}/scans/subfolder`, and `PUT /discovery/scan/folder/{folderId}` operations.

Parameter	Type	Description
folder	<a href="#">FolderValuesResult</a>	This class provides the result of the create or update operation of an object folder.

### PutScanConfigurationResult

This class returns the result of the create or update operation of the scan configuration. It is returned by the `PUT /discovery/scan/configuration` and `PUT /discovery/scan/configuration/folder/{folderId}` operations.

Parameter	Type	Description
rule	<a href="#">ScanConfigurationValuesResult</a>	This array returns the result and unique id of the scan configuration that was created or updated.

### PutTargetListResult

This class returns the result of the create or update operation of the target list. It is returned by the `PUT /discovery/target/list/folder/{folderId}` and `PUT /discovery/target/list` operations.

Parameter	Type	Description
rule	<a href="#">TargetListValuesResult</a>	This array returns the unique id of the target list and the unique id of the link between the target list and the member that was created or updated.

### ScanAssignment

This class provides the information on the scanners assigned to the scan. It is returned by the `GET /discovery/scan/{scanId}/assignment` operation.

Parameter	Type	Description
scannerId	integer	The scanner id of scanner.
ScannerName	string	The name of scanner the scanner.
assignmentId	integer	The assignment id of the link between the scanner and the scan job.
scheduleId	integer	The id of the schedule of the scan.
status	string	The status of the scan.
lastStatusUpdate	string	The last status update date.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ScanAttrs

This class provides the information that is necessary to create a new asset discovery scan. It is required by the `PUT /discovery/device/{deviceId}/scan` and `PUT /discovery/scan/folder/{folderId}` operations.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.

### ScanConfigurationArray

This class returns the number of scan configurations found in the specific location and one scan configuration information array per scan configuration. It is returned by the `GET /discovery/scan/configurations/folder/{folderId}` and `GET /discovery/scan/configurations` operations.

Parameter	Type	Description
total	integer	The total number of scan configurations found in the specified location.
configuration	Array[ <a href="#">ScanConfigurationListValues</a> ]	One array for each configuration.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ScanConfigurationAssign

This class provides the information on the scan configuration assigned to the scan. It is provided by the `GET /discovery/scan/{scanId}/configuration` operation.

Parameter	Type	Description
assignmentId	integer	The assignment id of the link between the scan and the scan configuration.
Id	integer	The id of the scan configuration.
name	string	The name of the scan configuration.

### ScanConfigurationAttrs

This class provides the information that is necessary to create a new scan configuration. It is required by the `PUT /discovery/scan/configuration` and `PUT /discovery/scan/configuration/folder/{folderId}` operations.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.

### ScanConfigurationListValues

This class contains one array for each scan configuration.

Parameter	Type	Description
ScanConfigurationList	array[ <a href="#">ScanConfigurationListValues</a> ]	One array for each scan configuration.

### ScanConfigurationProtocolArray

This class returns the protocols of the specified scan configuration. It is returned by the `GET /discovery/scan/configuration/{configurationId}/protocols` operation.

Parameter	Type	Description
protocols	array[ <a href="#">ScanConfigurationProtocolValues</a> ]	One array for each protocol of the scan configuration.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### ScanConfigurationProtocolValues

This class contains one array for each scan configuration protocol.

Parameter	Type	Description
ScanConfigurationProtocolValues	array[ <a href="#">ScanConfigurationProtocolValuesArray</a> ]	One array for each scan configuration protocol.

### ScanConfigurationProtocolValuesArray

This class returns the number of scan configuration protocols found for a scan configuration and one array for each protocol.

Parameter	Type	Description
assignmentId	integer	The unique BCM database identifier of the link between the scan configuration and the protocol.
isActive	integer	Indicates if the protocol is active or not.
EnumValue	string	Name of the protocol
AdminLogin	string	List of the login and encrypted password for the protocol.
Description	string	Description of the protocol.
IsAuth	integer	Indicates if the assignment is authenticated.
HasEncPasswd	integer	Indicates if the assignment has an encrypted password.

### ScanConfigurationValuesArray

This class returns the information that is available for a scan configuration.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the scan configuration.
name	string	The name of the scan configuration.

Parameter	Type	Description
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

### ScanConfigurationValuesResult

This class returns the result of the create or update operation of the scan configuration.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the scan configuration.

### ScanResult

This class returns the result of the scan.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the scan.
scheduleId	integer	The unique BCM database identifier of the scan's schedule.

### SNMPCredentialAttrs

This class provides the information that is necessary to access the SNMP protocol of a scan configuration. It is required by the `PUT /discovery/scan/configuration/{configurationId}/protocols` operation.

Parameter	Type	Description
community	string	Indicates the community (default public).
user	string	Indicates the name of the user.
authalgo	string	Indicates the type of authentication to use; either MD5 or SHA1 (default MD5).
authpass	string	Indicates the password to be used for the authentication.
privacyalgo	string	Indicates the type of privacy to use; either AES or DES (default AES).

Parameter	Type	Description
privacypass	string	Indicates the password to be used for the defined privacy.

### SNMPProtocolAttrs

This class provides the SNMP protocol status and credentials of a scan configuration. It is required by the `PUT /discovery/scan/configuration/{configurationId}/protocols` operation.

Parameter	Type	Description
activated	integer	Indicates if the protocol is activated.
credentials	array[ <a href="#">SNMPCredentialAttrs</a> ]	Array containing SNMP credentials.

### StatusValues

This class provides information about different assignment statuses.

Parameter	Type	Description
_STATUS_ASSIGNPAUSED_	string	The assignment order of the object to the target has been created but is currently either on hold or has not yet been ordered to be sent, because the object assignment has not been activated.
_STATUS_UNASSIGNEDPAUSED_	string	The unassignment request was defined on the Console and was about to be sent to the device, but the send action has been paused or the object has not yet been activated.
_STATUS_ASSIGNWAITING_	string	The assignment of the object to the target has been activated and is waiting to be sent. If the target object is <code>&amp;#147;unknown&amp;#148;</code> , the assignment remains in this status until the target becomes <code>&amp;#147;known&amp;#148;</code> .
_STATUS_UPDATEWAITING_	string	This status only indicates that the update information is waiting to be sent to the target.
_STATUS_REASSIGNWAITING_	string	An object that has already been executed on a specific target can be specified to execute on the same target again. This status indicates that the reassignment of the object to the target has been carried out on the Console and it is currently waiting to be sent to the target again.
_STATUS_UNASSIGNWAITING_	string	The unassignment request is defined and is waiting to be sent.

### TargetAssignIds

This class provides the unique BCM database identifier of an assignment link between the target list and members. If the operation is an update, the id already exists. Otherwise, a new id is created.



Parameter	Type	Description
Id	integer	The unique BCM database identifier of the assignment between the target list and the members. If the operation is an update, the id already exists. Otherwise, a new id is created.
targetMembers	[ <a href="#">MemberAttrs</a> ]	The list of members.

### TargetListArray

This class returns the number of target lists found in the specific location and one target list information array per target list. It is returned by the `GET /discovery/target/lists/folder/{folderId}` and `GET /discovery/target/lists` operations.

Parameter	Type	Description
total	integer	The total number of target lists found in the specified location.
targetlists	Array[ <a href="#">TargetListValues</a> ]	One array for each target list.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### TargetListAttrs

This class provides the information that is necessary to create a new target list. It is required by the `PUT /discovery/target/list/folder/{folderId}` and `PUT /discovery/target/list` operations.

Parameter	Type	Description
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
notes	string	This free text field can contain additional information concerning the selected object.

Parameter	Type	Description
targetMembers	array[ <a href="#">MemberAttrs</a> ]	Enter the name of the device to be added to the list. The name may be entered either as its short or long network name, for example, <i>scotty</i> or <i>scotty.enterprise.com</i> or as its IP address, for example, <i>159.124.5.10</i> or <i>2001:db8:85a3::8a2e:370:7334</i> , or as a comma separated list of names and/or ranges, for example, <i>scotty,192.168.4.45-192.168.4.47,2001:db8:85a3::8a2e:370:7334</i> which includes machines <i>scotty.enterprise.com</i> , <i>2001:db8:85a3::8a2e:370:7334</i> , <i>192.168.4.45</i> , <i>192.168.4.46</i> and <i>192.168.4.47</i> . A range may also be entered as CIDR notation in the form of <i>192.9.205.22/18</i> or <i>2001:db8:85a3::8a2e:370:152/896</i> . Be aware that the address <i>x.x.x.255</i> , being a broadcast IP address cannot be scanned and therefore cannot be included in any target range.

### TargetListMemberArray

This class returns the number of target list members found in the specific location and one member information array per list. It is returned by the `GET /discovery/target/list/{targetListId}/members` operation.

Parameter	Type	Description
total	integer	The total number of operational rules found in the specified location.
members	array[ <a href="#">TargetListMemberValues</a> ]	One array for each member of the target list.
errorCode	integer	The error code returned by the operation if the operation failed or <i>0</i> for a successful operation.

### TargetListMemberValues

This class returns one array for each member of the target list.

Parameter	Type	Description
targetListMemberValues	array[ <a href="#">TargetListMemberValuesArray</a> ]	One array for each member of the target list.

### TargetListMemberValuesArray

This class returns the information that is available for a target list member.

Parameter	Type	Description
assignmentId	integer	The unique BCM database identifier of the link between the target list and the members.
name	string	The name of the object under which it is known in BCM. This name may be any combination of characters.
excludedMember	string	The members excluded from the target list.
isGroup	boolean	Always false in this case.

### TargetListScanAssignArray

This class provides the information on the target lists assigned to the scan. It is returned by the `GET /discovery/scan/{scanId}/target/lists` operation.

Parameter	Type	Description
total	integer	The total number of target lists assigned to the scan.
members	array[ <a href="#">TargetListScanAssignValues</a> ]	One array for each target list assigned to the scan.
errorCode	integer	The error code returned by the operation if the operation failed or 0 for a successful operation.

### TargetListScanAssignValues

This class returns one array for each scan assigned to the target list.

Parameter	Type	Description
TargetListScanAssignValues	array[ <a href="#">TargetListScanAssignValuesArray</a> ]	One array for each scan assigned to the target list.

### TargetListScanAssignValuesArray

This class returns the information available for each scan assigned to a target list.

Parameter	Type	Description
assignmentId	integer	The assignment id of the link between the scan and the target list.
Id	integer	The id of the target list.
name	string	The name of the target list.

### TargetListValues

This class returns one array for each target list.

Parameter	Type	Description
TargetList	array[ <a href="#">TargetListValuesValues</a> ]	One array for each target list.

### TargetListValuesArray

This class returns the information that is available for a target list.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the target list.
name	string	The name of the target list.
notes	string	This free text field can contain additional information concerning the selected object.
createDate	string	The date and time at which the object was created in the BCM database.
updateDate	string	The date and time at which the object was last modified in the BCM database.

### TargetListValuesResult

This class returns the result of the create or update operation of the target list.

Parameter	Type	Description
Id	integer	The unique BCM database identifier of the target list.
assignId	integer	The unique BCM database identifier of the link between the target list and the members.

## Practical Examples

In this section you can see some practical examples on how to use the REST operations with the BMC Client Management database.

- [Creating a new custom inventory object type, adding it to the custom inventory of devices and filling it with information.](#)
- [Creating and Using a new Quick Link Operational Rule with a User.](#)
- [Creating a new Operational Rule in a new Folder.](#)
- [Creating a Software Distribution Rule.](#)
- [Creating a Dependency between Operational Rules.](#)
- [Assigning a Specific Schedule to an Operational Rule.](#)

## Creating and Using a new Custom Inventory Object Type

This example shows you how to create a new custom inventory object type in the BMC Client Management database and how to use it. This requires the following steps:

1. Check if the necessary custom inventory object type already exists in the BCM database
2. Create the new custom inventory object type
3. Create attributes for the new custom inventory object type
4. Add the new object type to the custom inventory of a device
5. Set the attributes for the custom inventory object type of the device

For this example we will create a new object type *Location* in which the respective data of a device are recorded.

1. Open the `/objects` section.

2. To enumerate the existing object types for the custom inventory class use the `GET /object` operation to ensure that the object type does not yet exist:

```
GET 127.0.0.1:1611/api/1/objects?class=_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_
```



This will return the list object types that already exist, for example, the default ones imported via the predefined objects: **Windows MSS** (object type Id 6109) and **Windows Security Settings** (object type Id 6110).

```
{
  "types": [
    {
      "id": 6109,
      "name": "Windows MSS",
      "class": "_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_",
      "smallIcon": null,
      "largeIcon": null,
      "parentType": null,
      "availableForSearch": false,
      "availableForQuery": false,
      "isGroup": false,
      "checkSecurity": false
    },
    {
      "id": 6110,
      "name": "Windows Security Settings",
      "class": "_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_",
      "smallIcon": null,
      "largeIcon": null,
      "parentType": null,
      "availableForSearch": false,
      "availableForQuery": false,
      "isGroup": false,
      "checkSecurity": false
    }
  ]
}
```

3. Open the `/object` section to create the new object type.

4. To create the new object type use the `PUT /object` operation:

```
PUT 127.0.0.1:1611/api/1/object?class=_DB_OBJECTTYPCLASS_CUSTOMINVENTORY_& type=Location
```



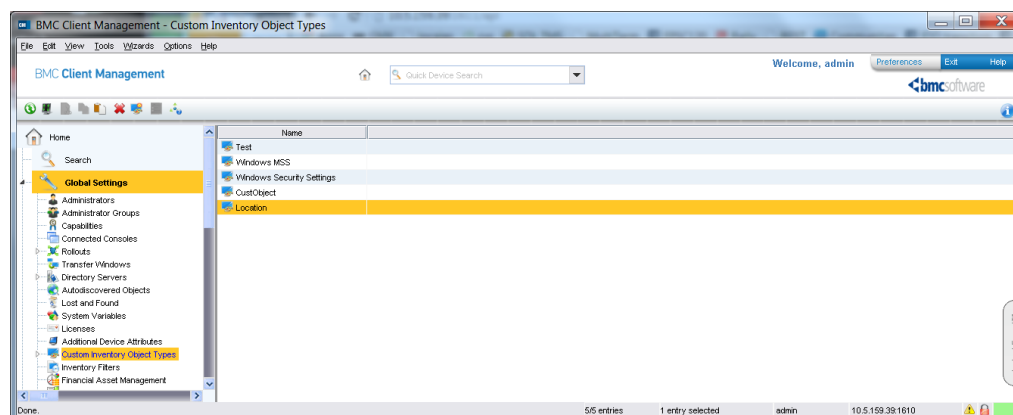
This operation returns the `PutObjectResult` class, which contains the object ID of the new object type and the corresponding error code `0` to indicate a successful operation: for our example the object type Id of *Location* is 6111.

```
{
  "Values": [
    "ObjectTypeId": 6111
  ],
  "ErrorCode": 0
}
```

```
}

```

You can also verify this in the BCM console in the **Global Settings > Custom Inventory Object Types** view:



5. Now that the new object type exists its attributes must be created, for this we will use the operation `PUT /object/{objectId}/attr:`

```
PUT 127.0.0.1:1611/api/1/object/6111/attr
```

The body must contain the definitions for the new attribute:

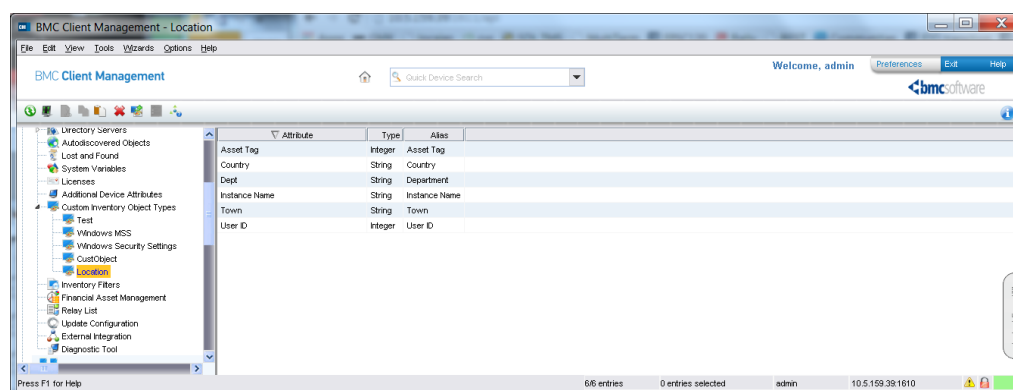
```
{ "AttrName": "Country",
  "AttrType": "String" }
```



This operation returns the `PutObjectAttrResult` class, which contains all available information about the new object type and its attribute and again the corresponding error code:

```
{
  "ObjectTypeId": 6111,
  "AttrName": "Country",
  "AttrType": "String",
  "ErrorCode": 0
}
```

You can also verify this in the BCM console in the **Global Settings > Custom Inventory Object Types > Location** view:



6. Repeat the last step for all attributes that the object type is to have:

- *Town*: string
- *Department*: string
- *Asset Tag*: integer

- *UserID*: integer



The new object type is now created with all its attributes.

7. Then you need to verify that all attributes are properly created and at the same time find their respective IDs. For this we will use the operation `GET /object/{objectId}/attrs`.

`GET 127.0.0.1:1611/api/1/object/6111/attrs`



This operation returns a list of all attributes of the new type, the `ObjectAttributes` class. The first three list items are the object type default attributes `_DB_ATTR_LASTINSERT_`, `_DB_ATTR_LASTUPDATE_` and `InstanceName`. All following list items are the newly created attributes and their values:

```
{
  "attrs":[
    {
      ....
    }
    {
      "name":"Country",
      "column":"ATTR_597033",
      "type":"String",
      "enumName":null,
      "units":null,
      "readOnly":false,
      "displayOrder":2,
      "displayInList":false,
      "needTranslation":false,
      "valueCheckRegex":null,
      "id":597075
    }
    {
      "name":"Town",
      "column":"ATTR_597034",
      "type":"String",
      "enumName":null,
      "units":null,
      "readOnly":false,
      "displayOrder":2,
      "displayInList":false,
      "needTranslation":false,
      "valueCheckRegex":null,
      "id":597076
    }
    {
      "name":"Department",
      "column":"ATTR_597035",
      "type":"String",
      "enumName":null,
      "units":null,
      "readOnly":false,
      "displayOrder":2,
      "displayInList":false,
      "needTranslation":false,
      "valueCheckRegex":null,
      "id":597077
    }
    {
      "name":"Asset Tag",
      "column":"ATTR_597036",
      "type":"Integer",
      "enumName":null,
      "units":null,
      "readOnly":false,
      "displayOrder":2,
      "displayInList":false,
      "needTranslation":false,
      "valueCheckRegex":null,
      "id":597078
    }
  ]
}
```

```
{
  "name": "User ID",
  "column": "ATTR_597037",
  "type": "Integer",
  "enumName": null,
  "units": null,
  "readOnly": false,
  "displayOrder": 2,
  "displayInList": false,
  "needTranslation": false,
  "valueCheckRegex": null,
  "id": 597079
}
```

8. Now the object type must be added to the custom inventory of the target devices and filled in with the data. For this open the `/device` section.

9. To add the new *Location* object type to the custom inventory we need to use the operation `PUT /device/{deviceid}/object/{objectid}/inst`. We will use the master (id=1000) as the target device:

```
PUT 127.0.0.1:1611/api/1/device/1000/object/6111/inst
```



This operation returns the object id of the new instance, 1000 in this case as well as the corresponding error code.

```
{
  "ObjectId": 1000,
  "ErrorCode": 0
}
```

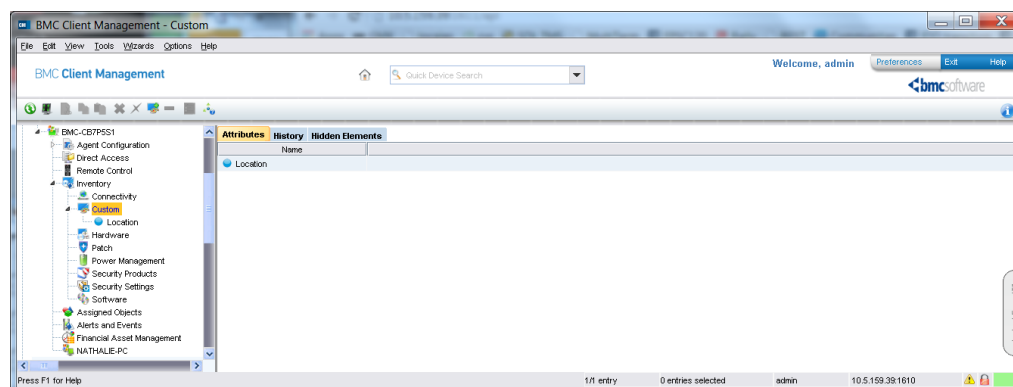


#### Note

Keep note of this object id, you will need it for later operations.

The object type is now added to the custom inventory. You can repeat the preceding step if you need more than one instance of the object type, for example if the device is a laptop and its user is traveling regularly between specific offices.

You can also verify this in the BCM console in the **Device Topology > Master > Inventory > Custom** view:



10. To fill in the data for the new object type in the custom inventory we need to use the operation `PUT /device/{deviceid}/object/{objectid}/inst/{instid}/attrs`:

```
PUT 127.0.0.1:1611/api/1/device/1000/object/6111/inst/1000/attrs
```

The body must contain the attribute values of the instance:

```
{
  "InstanceName": "Location Weekdays",
  "Country": "France",
  "Town": "Nantes",
}
```



```

"Department": "Accounting 157.EMEA",
"Asset Tag": "548925",
"User ID": "678"
}

```



### Note

Make sure you enter the names of the attributes exactly as you defined it in step 5, otherwise the values will not be set.



This operation returns the `DeviceOperationResult` displaying the data of the new attribute values as well as the corresponding error code.

```

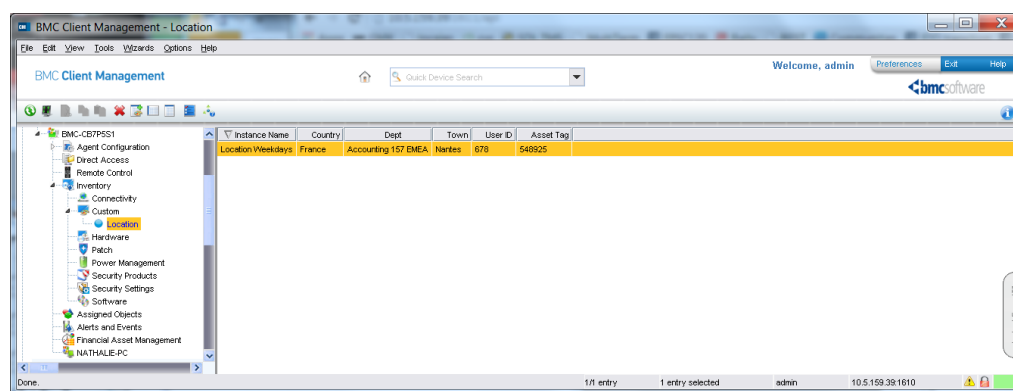
{
  "ParentType": "_DB_OBJECTTYPE_DEVICE_",
  "ParentId": 1000,
  "ObjectTypeId": 6106,
  "ObjectId": 1000,
  "ObjectAttrs": {
    "InstanceName": "Location Weekdays",
    "Country": "France",
    "Town": "Nantes",
    "Department": "Accounting 157.EMEA",
    "Asset Tag": "548925",
    "User ID": "678"
  },
  "ErrorCode": 0
}

```

whereby:

- `ParentType` is the type of the object, in this case it is a device
- `ParentId` is the database ID of device
- `ObjectTypeId` is the database ID of the *Location* object
- `ObjectId` is the database ID of the *Location* object instance

To verify this in the BCM console return to the **Device Topology > Master > Inventory > Custom** view and select the *Location*:



11. If you have more than once instance repeat this step for all custom inventory instances of the object.



You have now created the new *Location* custom inventory object type with all its attributes, assigned it with one instance to the custom inventory of the master and filled the attributes with their vales.

## Creating and Using a new Quick Link Operational Rule

This example shows you how to create a new folder under the root folder and add a quick link rule to it. This requires the following steps:

1. Check if the folder already exists
2. Create the new folder
3. Create quick link rule

For this example we will create a new object type *Location* in which the respective data of a device are recorded.

1. Open the `/oprule` section.

2. To enumerate the existing operational rule folders use the `GET /oprule/subfolders` operation to ensure that the new folder to be created does not yet exist. Set the `brief` parameter to `true` to only list the folders without any other information:

```
GET 127.0.0.1:1611/api/1//oprule/subfolders?
brief=true&sort=name&asc=true&hierarchy=false&count=100
```



This will return the `OperationalRuleFolderArray` class, which lists all folders that already exist under the main operational rules folder, for example, all folders imported via the out of the box objects.

```
{
  ...
}
```

3. To create the new folder use the `PUT oprule/subfolder` operation:

```
PUT 127.0.0.1:1611/api/1/oprule/subfolder
```

The body must contain the folder information, for example:

```
{
  "name": "My New Test Rules",
  "notes": "New rules folder for testing purposes"
}
```



This operation returns the `PutRuleFolderResult` class, which contains the object ID of the new folder.

```
{
  "folder":{
    "id":136
  }
}
```

4. Now that the new folder exists we will create a new operational rule of type **Quick Link**, for this we will use the operation `PUT /oprule/folder/{folderId}/rule`:

```
PUT 127.0.0.1:1611/api/1/oprule/folder/136/rule
```

The body must contain the definitions for the new rule:

```
{
  "name": "My quick link rule",
  "type": "Quick Link",
  "url": "http://www.mycompany.com"
  "notes": "A rule to make some tests",
  "kioskIcon": "or/Kiosk/globe.png"
}
```




This operation returns the `PutRuleResult` class, which contains the ID of the new rule.

```
{
  "rule":{
    "id":1005
  }
}
```

5. Now, that the rule is defined it must be assigned to a target. For our example we will assign it to a user group called *All Users*. For this we first need to find the ID of this group via the `GET /user/subgroups` operation in the `group` group.

```
PUT 127.0.0.1:1611/api/1/user/subgroups?
brief=false&sort=name&asc=true&hierarchy=false&count=100
```

 This operation returns the `GroupUsers` class, which contains the ID of all subgroups.

```
{
  "groups": [
    {
      "id": 178,
      "name": "All Users",
      "notes": "",
      "createDate": 1361527328,
      "updateDate": 1361527357
    },
    {
      "id": 242,
      "name": "Out-of-the-Box#####",
      "notes": "",
      "createDate": 1373360814,
      "updateDate": 1379937481
    }
  ]
}
```


The ID of the group we are looking for is 178.

6. The assignment between the rule and the user group is executed via the `PUT /oprule/rule/{ruleId}/user/group/{groupId}` operation. To immediately activate the rule assignment leave the `activation` parameter as it is, otherwise set it to `manual`:




```
PUT 127.0.0.1:1611/api/1/oprule/rule/1006/device/1000?activation>manual
```

The body must contain the definitions for the assignment, for example:

```
{
  "installAsUser": "true",
  "modeMyApps": "true",
  "notOnlyPrimaryUser": "false",
  "userAssignPolicy": "_ASSIGN_IMMEDIATELY_"
}
```

 This operation returns the `AssignmentDetails` class, which contains the ID of the assignment.

```
{
  "assignment": {
    "id": 1001
  }
}
```

You should now also see that the blue icon of the BCM agent in the systray changed from  to  with the **New** banner to indicate that a new rule is available in **MyApps**. If you double-click  a browser window opens and you should see the rule on the



You have now created the new quick link rule *My quick link rule* in a new operational rules folder and made it available on the **MyApps** application kiosk.

## Creating a new Operational Rule in a new Folder

This example shows you how to create a new operational rule subfolder and operational rules in the BMC Client Management database and how to use them. This requires the following steps:

1. Create the new operational rule folder
2. Create the new operational rule

3. Find the step to use with the rule and assign them
4. Assign the rule to the target device

For this example we will create a new subfolder, *My Second New Test Rules Folder*, under an existing folder, *My New Test Rules* and create a new operational rule, *My Regular Operational Rule*. Then we will assign it to the master device.

1. Open the `/oprules` section.
2. As we just created the folder *My New Test Rules* in the last example (ID 136), we know that there are no subfolders in it yet and therefore we can directly create our new subfolder via the `PUT /oprule/folder/{folderId}/subfolder` operation:

```
GET 127.0.0.1:1611/api/1/oprule/folder/136/subfolder
```

The body must contain the definitions for the new attribute:

```
{
  "name": "My Second New Test Rules Folder"
}
```



This operation returns the `OperationalRuleFolderValues` class, which contains the object ID of the new folder and its available information.

```
{
  "folder": {
    "id": 137
  }
}
```

3. To verify that the new folder was correctly created we will call the `GET /oprule/folder/{folderId}/subfolders` operation:

```
PUT 127.0.0.1:1611/api/1/oprule/folder/136/subfolders?
brief=true&sort=name&asc=true&hierarchy=false&count=100
```



This operation returns the `OperationalRuleFolderArray` class, which contains the list of all subfolders, in our case only one.

```
{
  "total": 1,
  "folders": [
    {
      "id": 137,
      "name": "My Second New Test Rules Folder",
      "notes": "",
      "createDate": 1380730909,
      "updateDate": 1380731097
    }
  ]
}
```

4. Now we can create our operational rule; for this we will use the operation `PUT /oprule/folder/{folderId}/rule`:

```
PUT 127.0.0.1:1611/api/1/oprule/folder/137/rule
```

The body must contain the name for the new rule:

```
{
  "name": "My Regular Operational Rule",
  "type": "Operational Rule"
}
```



This operation returns the `PutRuleResult` class, which contains the ID of the new rule.

```
{
  "rule": {
    "id": 1005
  }
}
```

5. Now we must find the ID of the operational rule step that we want to assign to our rule via the `GET /oprule/steps` operation. The list can be pre-filtered using the `query` parameter, for example, to only return all steps that contain the expression `wait` in the English language locale.

```
PUT 127.0.0.1:1611/api/1/oprule/steps?query=wait&language=English
```



This operation returns the `StepList` class, which lists all steps that match the query.

```
{
  "steps": [
    {
      "id": 1073,
      "name": "_DB_STEPNAME_FILESTOREMANUALUPLOAD_",
      "class": "_DB_STEPCLASS_AGENTCONFIG_",
      "notes": "_DB_STEPNOTE_FILESTOREMANUALUPLOAD_"
    },
    {
      "id": 1248,
      "name": "_DB_STEPNAME_WAITPROCESS_",
      "class": "_DB_STEPCLASS_PROCESSTMGT_",
      "notes": "_DB_STEPNOTE_WAITPROCESS_"
    },
    {
      "id": 1249,
      "name": "_DB_STEPNAME_WAIT_",
      "class": "_DB_STEPCLASS_TOOLS_",
      "notes": "_DB_STEPNOTE_WAIT_"
    },
    {
      "id": 1247,
      "name": "_DB_STEPNAME_WAITNBDBAYS_",
      "class": "_DB_STEPCLASS_TOOLS_",
      "notes": "_DB_STEPNOTE_WAITNBDBAYS_"
    }
  ]
}
```

6. For our example we will use the step with ID 1249 (`_DB_STEPNAME_WAIT_`: **Wait**). To be able to add an instance of it to a rule we first need to find the IDs of all its parameters via the `GET /oprule/step/{stepTypeId}` operation.

```
PUT 127.0.0.1:1611/api/1/oprule/oprule/step/1249
```



This operation returns the list of step parameters with their types and IDs ( `StepDetails` class).

```
{
  "step": {
    "id": 1249,
    "params": [
      {
        "id": 2214,
        "label": "_DB_STEPPARAM_WAITNBSECONDS_",
        "name": "WaitTime",
        "type": "Integer"
      }
    ]
  }
}
```

7. Now to add an instance of this step to the rule *My Regular Operational Rule* (ID=1005) we need to use the operation `PUT /oprule/rule/{ruleId}/step/{stepTypeId}`:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/1005/step/1249
```

The body must contain the parameter value for the new step instance:

```
{
  "params": [
    {
      "id": 2214,
      "value": "10"
    }
  ]
}
```

}



This operation returns the `PutRuleStepAssignmentResult` class, which contains the ID of the assignment between the step and the rule.

```
{
  "assignment": {
    "id": 1003
  }
}
```

8. To get more information about this new assignment use the `GET /oprule/rule/{ruleId}/step/assignment/{assignmentId}` operation:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/1005/step/assignment/1003
```



This operation returns the `StepAssignment` class, which contains the ID of the assignment between the step and the rule.

```
{
  "assignment": {
    "id": 1003,
    "position": 1,
    "step": {
      "id": 1249,
      "params": [
        {
          "id": 2214,
          "label": "_DB_STEPPARAM_WAITNBSECONDS_",
          "name": "WaitTime",
          "type": "Integer",
          "value": "10"
        }
      ]
    }
  }
}
```

9. Now, that the rule is defined it must be assigned to a target, for example to the master device (ID=1000), via the operation `PUT /oprule/rule/{ruleId}/device/{deviceId}`. To immediately activate the rule assignment leave the `activation` parameter as it is, otherwise set it to `manual`:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/1005/device/1000?activation>manual
```



This operation returns the `AssignmentDetails` class, which contains the ID of the assignment.

```
{
  "assignment": {
    "id": 1006
  }
}
```

10. To get the details of this assignment use the `GET /oprule/rule/device/assignment/{assignmentId}` operation.

```
PUT 127.0.0.1:1611/api/1/oprule/rule/device/assignment/1006
```



This operation returns the `DeviceAssignmentAttrsObject` class, which contains the details of the assignment:

```
{
  "assignment": {
    "id": 1006,
    "device": {
      "id": 1000,
      "group": {
        "id": 0
      }
    },
    "schedule": {
      "id": 1015
    }
  }
}
```

```

    },
    "status": "_STATUS_ASSIGNPAUSED_",
    "transportMode": "",
    "specialCase": false,
    "bypassTransferWindows": false,
    "wakeupDevices": false,
    "uploadStatus": true,
    "executeWhileFails": false,
    "uploadStatusEveryExec": false,
    "runAsCurrentUser": false,
    "executionError": {
        "errorStepNumber": 0,
        "errorType": "",
        "errorCode": 0,
        "errorText": ""
    }
  }
}

```



You have now created the new operational rule *My Regular Operational Rule*, which will wait 10 seconds before allowing to continue with any other rules that might be dependent on it, and assigned it to the master device. The rule is, however, not yet activated, that is, it will not yet be sent to the target for execution.

## Creating Software Distribution Rule

In this example we will create a new software distribution rule in the already existing folder *My New Test Rules*.



### Note

Be aware, that you need to have a package available, for example one of the upgrade packages or another package created in BMC Client Management by default, such as the `NmapFiles.cst` package.

This requires the following steps:

1. Create the new software distribution rule
2. Find the step to use with the rule and assign it
3. Find the package and assign it
4. Assign the rule to the target device

In this example we will create a second operational rule, *My Software Distribution Rule*, in the same folder as the *My Regular Operational Rule* rule, the *My New Test Rules* folder. This rule will check the operating system of the targets and if it matches, it will install a software package. This rule can be installed with or without automatic activation.

1. Open the `/oprules` section.
2. To verify that we have the correct ID of folder in which to create the new rule call the `GET /oprule/folder/{folderId}/subfolders` operation, in this case with the `brief` parameter set to `false` to return further information about the folder, such as the name, and the `hierarchy` parameter set to `true` to also display all rules of this folder:

```

PUT 127.0.0.1:1611/api/1/oprule/folder/136/subfolders?
brief=false&sort=name&asc=true&hierarchy=true&count=100

```



This operation returns the `OperationalRuleFolderArray` class, which contains the list of all subfolders, in our case only one, and the quick link rule that was created in a previous example.

```

{
  "total": 2,
  "folders": [
    {
      "id": 137,
      "name": "My Second New Test Rules Folder",
      "notes": "",
      "createDate": 1381153058
    }
  ]
}

```

```

    "rules": [
      {
        "id": 1070,
        "name": "My quick link rule",
        "notes": "A rule to make some tests",
        "createDate": 1381153287,
        "updateDate": 1381153287
      }
    ]
  }
}

```

To ensure that the *My Second New Test Rules Folder* folder, is the folder that already contains the *My Regular Operational Rule* rule, you could re-execute this operation with the ID 137. The result should in this case then display an empty folder array and one object in the rules array.

- Now we can create our operational rule; for this we will use the operation `PUT /oprule/folder/{folderId}/rule:`

```
PUT 127.0.0.1:1611/api/1/oprule/folder/137/rule
```

The body must contain the name for the new rule:

```

{
  "name": "My Software Distribution Rule",
  "type": "Operational Rule"
}

```



#### Note

Operational rules of type **Software Distribution** are initially also created as rules of the standard type **Operational Rule**. Their type will automatically be changed to **Software Distribution** once the package to distribute is added.



This operation returns the `PutRuleResult` class, which contains the ID of the new rule.

```

{
  "rule": {
    "id": 1006
  }
}

```

- Now we must find the IDs of the operational rule steps that we want to assign to our rules via the `GET /oprule/steps` operation. The list can be pre-filtered using the `query` parameter to only return all steps that contain the expression *operating system*.

```
PUT 127.0.0.1:1611/api/1/oprule/steps?query='operating system'&language=English
```



This operation returns the `StepList` class, which lists all steps that match the query in the English language locale.

```

{
  "steps": [
    {
      "id": 1026,
      "name": "_DB_STEPNAME_CHECKNEWDEVICES_",
      "class": "_DB_STEPCLASS_MASTERSTEPS_",
      "notes": "_DB_STEPNOTE_CHECKNEWDEVICES_"
    },
    {
      "id": 1029,
      "name": "_DB_STEPNAME_CHECKOPERATINGSYSTEM_",
      "class": "_DB_STEPCLASS_MONITORING_",
      "notes": "_DB_STEPNOTE_CHECKOPERATINGSYSTEM_"
    },
    {
      "id": 1286,
      "name": "_DB_STEPNAME_PMGMTMACDEPLOYPATCHES_",
      "class": "_DB_STEPCLASS_PATCHMANAGEMENT_",
      "notes": "_DB_STEPNOTE_PMGMTMACDEPLOYPATCHES_"
    }
  ]
}

```



```

    },
    {
      "id": 1132,
      "name": "_DB_STEPNAME_REBOOT_",
      "class": "_DB_STEPCLASS_TOOLS_",
      "notes": "_DB_STEPNOTE_REBOOT_"
    },
    {
      "id": 1195,
      "name": "_DB_STEPNAME_SHUTDOWN_",
      "class": "_DB_STEPCLASS_TOOLS_",
      "notes": "_DB_STEPNOTE_SHUTDOWN_"
    },
    {
      "id": 1293,
      "name": "_DB_STEPNAME_CHECKOPERATINGSYSTEM2_",
      "class": "_DB_STEPCLASS_TOOLS_",
      "notes": "_DB_STEPNOTE_CHECKOPERATINGSYSTEM2_"
    },
    {
      "id": 1109,
      "name": "_DB_STEPNAME_MSGBOXLOCALIZED_",
      "class": "_DB_STEPCLASS_USERDIALOG_",
      "notes": "_DB_STEPNOTE_MSGBOXLOCALIZED_"
    }
  ]
}

```

5. For our example we will use the step with ID 1293 (**\_DB\_STEPNAME\_CHECKOPERATINGSYSTEM2\_**: **Check Operating System**). To be able to add an instance of it to a rule we first need to find the IDs of all its parameters via the `GET /oprule/step/{stepTypeId}` operation.

PUT 127.0.0.1:1611/api/1/oprule/oprule/step/1293



This operation returns the list of step parameters with their types and IDs ( `StepDetails` class).

```

{
  "step": {
    "id": 1293,
    "params": [
      {
        "id": 2619,
        "label": "_DB_STEPPARAM_WIN95_",
        "name": "Windows95",
        "type": "Boolean"
      },
      {
        "id": 2620,
        "label": "_DB_STEPPARAM_WIN98_",
        "name": "Windows98",
        "type": "Boolean"
      },
      ....
      {
        "id": 2655,
        "label": "_DB_STEPPARAM_SOLARIS_",
        "name": "Solaris",
        "type": "Boolean"
      },
      {
        "id": 2656,
        "label": "_DB_STEPPARAM_MACOSX_",
        "name": "MacOSX",
        "type": "Boolean"
      }
    ]
  }
}

```

6. Now to add an instance of this step to the rule *My Regular Operational Rule* (ID=1006) we need to use the operation `PUT /oprule/rule/{ruleId}/step/{stepTypeId}`:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/1006/step/1293
```

The body must contain the parameter definitions for the step instance. The definition is to be defined as such: if the operating system is *not* Solaris or Mac OS, the package is to be installed. You could also do it the other way round, but in this case a very large number of parameters need to be defined.

```
{
  "stopCondition": "_DB_STEPACTION_SUCCEED_",
  "params": [
    {
      "id": "2655",
      "value": "Yes"
    },
    {
      "id": "2656",
      "value": "Yes"
    }
  ]
}
```



This operation returns the `PutRuleStepAssignmentResult` class, which contains the ID of the assignment between the step and the rule.

```
{
  "assignment": {
    "id": 1380
  }
}
```

7. The next step is to assign the package. For this we first need to find it's ID via the operation `GET package/packages` of the `/group` group:

```
GET 127.0.0.1:1611/api/1/package/packages?brief=false&sort=name&asc=true&count=100
```



This operation returns the `Packages` class displaying the of all packages located directly under the main packages node, in this case there are 3.

```
{
  "packages": [
    {
      "id": 1007,
      "name": "DiagFiles.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1375352384,
      "updateDate": 1375352384
    },
    {
      "id": 1001,
      "name": "New Package.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1366033698,
      "updateDate": 1366033698
    },
    {
      "id": 1000,
      "name": "NmapFiles.cst",
      "notes": "",
      "type": "CUSTOM",
      "createDate": 1361527376,
      "updateDate": 1361527376
    }
  ]
}
```

8. Now that we know the ID of the package we can assign it via the `PUT /oprule/rule/{ruleId}/package/{packageId}` operation:

```
GET 127.0.0.1:1611/api/1/oprule/rule/1006/package/1000
```



This operation returns the `OperationResult` class displaying the result of the operation in the form of its error code.

```
{
  "errorCode": 0
}
```

9. Now, that the rule is defined it must be assigned to a target, for example to the master device (ID=1000), via the operation `PUT /oprule/rule/{ruleId}/device/{deviceId}`. To immediately activate the rule assignment leave the `activation` parameter as it is, otherwise set it to `manual`:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/1006/device/1000?activation=manual
```



This operation returns the `AssignmentDetails` class, which contains the ID of the assignment.

```
{
  "assignment": {
    "id": 1022
  }
}
```

10. To get the details of this assignment use the `GET /oprule/rule/device/assignment/{assignmentId}` operation.

```
PUT 127.0.0.1:1611/api/1/oprule/rule/device/assignment/1007
```



This operation returns the `DeviceAssignmentAttrsObject` class, which contains the details of the assignment:

```
{
  "assignment": {
    "id": 1022,
    "device": {
      "id": 0,
      "group": {
        "id": 0
      }
    },
    "schedule": {
      "id": 1267
    },
    "status": "_STATUS_ASSIGNPAUSED_",
    "transportMode": "",
    "specialCase": false,
    "bypassTransferWindows": false,
    "wakeupDevices": false,
    "uploadStatus": true,
    "executeWhileFails": false,
    "uploadStatusEveryExec": false,
    "runAsCurrentUser": false,
    "executionError": {
      "errorStepNumber": 0,
      "errorType": "",
      "errorCode": 0,
      "errorText": ""
    }
  }
}
```



You have now created the new software distribution rule *My Regular Operational Rule*, which will first check the operating system of its target and then, if it is the correct one, install the associated package on it. The rule is, however, not yet activated, that is, it will not yet be sent to the target for execution.

## Creating a Dependency between Operational Rules

This example shows you how to create a dependency between two operational rules. This requires the following steps:

1. Check if both rules are in the operational rule folder
2. Create the dependency
3. Activate both operational rules

For this example we will verify that both rules are located in folder *My Second New Test Rules Folder* and then create a dependency between them. Then we will activate both.

1. Open the `/oprules` section.

2. To verify that we have the correct ID of folder which contains both rules call the `GET /oprule/folder/{folderId}/rules` operation:

```
PUT 127.0.0.1:1611/api/1/oprule/folder/259/rules?brief=false&sort=name&asc=true&count=100
```



This operation returns the `OperationalRuleArray` class, which contains the list of all rules located in it. In our case the two rules that we created in the last two examples.

```
{
  "total": 2,
  "rules": [
    {
      "id": 1005,
      "name": "My Operational Rule",
      "type": "Operational Rule",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/Kiosk/oprule.png",
      "notes": "",
      "createDate": 1381158776
    },
    {
      "id": 1006,
      "name": "My Software Distribution Rule",
      "type": "Software Distribution",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/kiosk/oprules_pkg.png",
      "notes": "",
      "createDate": 1381153685,
      "updateDate": 1381157347
    }
  ]
}
```

3. To create the dependency between these two rules we will call the `PUT /oprule/rule/{ruleId}/dependency/{dependentRuleId}` operation:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/1006/dependency/1005
```



This operation returns the `OpRuleDependencyOrder` class, which contains the position number of the dependency that was just assigned.

```
{
  "dependency": {
    "position": 1
  }
}
```

4. The rules can now be activated via the `PUT /oprule/rule/device/assignment/{assignmentId}`:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/device/assignment/1022
```

The body must contain the name for the new rule:

```
{
  "status": "_STATUS_ASSIGNWAITING_"
}
```



This operation returns the `OperationResult` class, which returns the assignment update result in the form of an error code.

```
{
  "errorCode": 0
}
```



The two new operational rules are now dependent on each other in their execution and they are activated with the default schedule.

## Assigning a Specific Schedule to an Operational Rule

This example shows you how to assign another than the default schedule to an operational rule. This requires the following steps:

1. Find the ID of the operational rule
2. Find the schedule ID
3. Update the schedule
4. Update the rule-device assignment to apply the new schedule

1. Open the `/oprules` section.

2. Check the folder in which the rule is located for the correct rule ID via the `GET /oprule/folder/{folderId}/rules` operation:

```
PUT 127.0.0.1:1611/api/1/oprule/folder/259/rules?brief=false&sort=name&asc=true&count=100
```



This operation returns the `OperationalRuleArray` class, which contains the list of all rules located in it. In our case the two rules that we created in the last two examples.

```
{
  "total": 2,
  "rules": [
    {
      "id": 1005,
      "name": "My Operational Rule",
      "type": "Operational Rule",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/Kiosk/oprule.png",
      "notes": "",
      "createDate": 1381158776
    },
    {
      "id": 1006,
      "name": "My Software Distribution Rule",
      "type": "Software Distribution",
      "deploymentFromExternalIntegration": "Not available in integrated solutions",
      "kioskIcon": "or/kiosk/oprules_pkg.png",
      "notes": "",
      "createDate": 1381153685,
      "updateDate": 1381157347
    }
  ]
}
```

3. The next step is to find the assignment ID via the `GET /oprule/rule/{ruleId}/device/assignments` operation, because in our case we are looking for a device assignment:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/1005/device/assignments?brief=false
```



This operation returns the `DeviceAssignmentAttrsArray` class, which lists all devices and their assignment data that are assigned to this rule.

```
{
  "total": 2,
```

```

"assignments": [
  {
    "id": 1022,
    "device": {
      "id": 1000,
      "name": "BMC-CB7L5S1",
      "groupName": ""
    },
    "schedule": {
      "id": 1267
    },
    "status": "_STATUS_VERIFICATIONFAILED_",
    "transportMode": "Unicast",
    "enableTime": false,
    "isActive": true,
    "networkInstall": false,
    "executeWhileFails": false,
    "uploadStatusEveryExec": false,
    "runAsCurrentUser": false,
    "executionError": {
      "errorStepNumber": 1,
      "errorType": "_DB_OPRULEERRORTYPE_EXTERN_",
      "errorCode": 9504,
      "errorText": "_ERROR_OSNOTCOMPLIANT_"
    }
  },
  {
    "id": 1023,
    "device": {
      "id": 1001,
      "name": "NewTestDevice",
      "groupName": ""
    },
    "schedule": {
      "id": 1268
    },
    "status": "_STATUS_IMPOSSIBLESENDING_",
    "transportMode": "",
    "enableTime": false,
    "isActive": true,
    "networkInstall": false,
    "executeWhileFails": false,
    "uploadStatusEveryExec": false,
    "runAsCurrentUser": false,
    "executionError": {
      "errorStepNumber": 0,
      "errorType": "",
      "errorCode": 0,
      "errorText": ""
    }
  }
]
}

```

4. In the next step we need to get the device assignment details via the `GET /oprule/rule/device/assignment/{assignmentId}`:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/device/assignment/1022
```



This operation returns the `DeviceAssignmentAttrsObject` class, which returns the assignment data, amongst other the ID of the schedule, in our example here this is 1267.

```

{
  "assignment": {
    "id": 1022,
    "device": {
      "id": 0,
      "group": {
        "id": 0
      }
    }
  }
}

```

```

    },
    "schedule": {
      "id": 1267
    },
    "status": "_STATUS_ASSIGNPAUSED_",
    "transportMode": "",
    "specialCase": false,
    "bypassTransferWindows": false,
    "wakeupDevices": false,
    "uploadStatus": true,
    "executeWhileFails": false,
    "uploadStatusEveryExec": false,
    "runAsCurrentUser": false,
    "executionError": {
      "errorStepNumber": 0,
      "errorType": "",
      "errorCode": 0,
      "errorText": ""
    }
  }
}

```

5. In the next step we display the schedule details via the `GET /schedule/{scheduleId}` to verify which parameters require changing:

```
PUT 127.0.0.1:1611/api/1/schedule/1267
```



This operation returns the `ScheduleResult` class, which returns the schedule data.

```

{
  "schedule": {
    "id": 1267,
    "name": "My Operational Rule",
    "validity": {
      "execution": "immediate",
      "executionAt": 0,
      "termination": "runCount",
      "terminationRunCount": 1,
      "terminationAt": 0
    },
    "frequency": {
      "type": "schedule",
      "scheduleMonths": [
        ""
      ],
      "period": "1min",
      "periodAt": "",
      "periodFrom": "*:00",
      "periodTo": "1:00",
      "scheduleType": "daily",
      "cron": 0
    }
  }
}

```

The currently assigned schedule is the default schedule which defines immediate assignment and immediate execution, once only.

6. For our example we want to schedule the execution still only once, but not immediately but during the night, when a package installation will not interfere with the daily operations via the `PUT /schedule/{scheduleId}` operation:

```
PUT 127.0.0.1:1611/api/1/schedule/1267
```


The body must contain the data to update the schedule with:

```

{
  "schedule": {
    "id": 1267,
    "validity": {
      "termination": "never"
    }
  }
}

```

```
    },
    "frequency": {
      "type": "schedule",
      "scheduleType": "dayOfWeek",
      "scheduleDayOfWeek": [
        "1",
        "2",
        "3",
        "4",
        "5"
      ],
      "scheduleMonths": [
        "*"
      ],
      "period": "at",
      "periodAt": "3:00",
    }
  }
}
```

 This operation returns the `OperationResult` class, which returns update result in the form of an error code.


```
{
  "errorCode": 0
}
```

7. The schedule is now updated, however to apply it the rule-target assignment must be activated again via the `PUT /oprule/rule/device/assignment/{assignmentId}`:

```
PUT 127.0.0.1:1611/api/1/oprule/rule/device/assignment/1022
```

The body must contain the name for the new rule:

```
{
  "status": "_STATUS_ASSIGNWAITING_"
}
```

 This operation returns the `OperationResult` class, which returns the assignment update result in the form of an error code.

```
{
  "errorCode": 0
}
```



The operational rule now has a new schedule and will be executed on the target every weekday at 3 am until further notice.

## Appendix

This appendix provides you with useful information about the use of the REST operations in connection with the BMC Client Management database and its objects.

- List of main BCM database objects

### Most Common Object Types and their IDs

Following you can see a list with the most common object class types and their ID.

ID	ObjectTypeClass	ObjectName
100	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_DEVICE_



ID	ObjectTypeClass	ObjectTypeName
101	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_DEVICEGROUP_
102	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_ADMINISTRATOR_
103	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_ADMINISTRATORGROUP_
107	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_OPERATIONALRULE_
108	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_OPERATIONALRULEFOLDER_
109	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_QUERY_
110	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_QUERYFOLDER_
111	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_REPORT_
112	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_REPORTFOLDER_
113	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_TRANSFERWINDOW_
114	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_TRANSFERWINDOWFOLDER_
115	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_ROLLOUT_
118	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_ROLLOUTFOLDER_
132	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_DIRECTORYSERVER_
134	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_PACKAGE_
135	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_PACKAGEFOLDER_
519	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_USER_
520	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_USERGROUP_
528	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_PATCHGROUP_
529	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_PATCHGROUPFOLDER_
554	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_COMPLIANCERULE_
555	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_COMPLIANCERULEFOLDER_
583	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_FAM_
565	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_APPLICATIONLISTS_
566	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_APPLICATIONLISTSFOLDER_
573	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_LICENSEDAPPLICATIONS_

ID	ObjectTypeClass	ObjectTypeName
574	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_LICENSEDAPPLICATIONSFOLDER_
593	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_PATCHJOB_
594	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_PATCHJOBFOLDER_
597	_DB_OBJECTTYPCCLASS_BASIC_	_DB_OBJECTTYPE_EXTINTEGRATION_
600	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	AuditPolicy
601	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	DevicePolicy
602	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	DomainPolicy
603	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	AccountPolicy
604	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	LogFilePolicy
605	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	LogonPolicy
614	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	FirewallInfos
616	_DB_OBJECTTYPCCLASS_SECURITYINVENTORY_	SharedResources
136	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Battery
137	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	BIOS
138	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Cache Memory
139	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	CDROM Drive
140	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Computer System
141	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Computer System Product
142	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Desktop Monitor
143	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Disk Drive
144	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Disk Partition
145	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Video Controller
146	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Floppy Drive
147	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Keyboard
148	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Logical Disk
149	_DB_OBJECTTYPCCLASS_HARDWAREINVENTORY_	Motherboard Device

ID	ObjectTypeClass	ObjectTypeName
150	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Network Adapter
151	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Parallel Port
152	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Physical Memory
153	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Pointing Device
154	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Printer
155	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Processor
158	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Operating System
615	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Network Adapter Configuration
800	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IAntivirus
801	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IFirewall
802	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IAntispyware
803	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IWebBrowser
804	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IInstantMessenger
805	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IP2p
806	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IHdEncryption
807	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IURLFiltering
808	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IAntiPhishing
809	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IHealthAgent
810	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IDeviceAccessControl
811	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IVpnClient
812	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IBackupClient
813	_DB_OBJECTTYPCLASS_SPMINVENTORY_	ISoftwareSuite
814	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IVirtualMachine
815	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IDataLossPrevention
816	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IDesktopSharing
817	_DB_OBJECTTYPCLASS_SPMINVENTORY_	ISystemManagement

ID	ObjectTypeClass	ObjectTypeName
818	_DB_OBJECTTYPCLASS_SPMINVENTORY_	IPatchManagement
6101	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Physical Memory Array
6102	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	Sound Device
6103	_DB_OBJECTTYPCLASS_HARDWAREINVENTORY_	OS Performance

