Bitdefender

GravityZone

API DOCUMENTATION

Bitdefender Control Center API Documentation

Publication date 2017.03.02

Legal Notice

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without written permission from an authorized representative of Bitdefender. The inclusion of brief quotations in reviews may be possible only with the mention of the quoted source. The content can not be modified in any way.

Warning and Disclaimer. This product and its documentation are protected by copyright. The information in this document is provided on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this document, the authors will not have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

This book contains links to third-party Websites that are not under the control of Bitdefender, therefore Bitdefender is not responsible for the content of any linked site. If you access a third-party website listed in this document, you will do so at your own risk. Bitdefender provides these links only as a convenience, and the inclusion of the link does not imply that Bitdefender endorses or accepts any responsibility for the content of the third-party site.

Trademarks. Trademark names may appear in this book. All registered and unregistered trademarks in this document are the sole property of their respective owners, and are respectfully acknowledged.

Table of Contents

1. GETTING STARTED

1.1. Introduction

Bitdefender Control Center APIs allow developers to automate business workflows.

The APIs are exposed using JSON-RPC 2.0 protocol specified here:

http://www.jsonrpc.org/specification.

Each API call targets a method and passes a set of parameters.

There are two types of parameters:

- required: MUST be always passed to the called method.
- optional: has a default value and can be omitted from the parameters list. Any optional parameter can be skipped, regardless its position in the parameters list.

1.2. API Requests

The API calls are performed as HTTP requests with JSON-RPC messages as payload. HTTP POST method MUST be used for each API call. Also, it is required that each HTTP request have the `Content-Type` header set to `application/json`.

Note
The API is limited to maximum 10 requests per second per API key. If this limit is exceeded, subsequent requests are rejected and 429 HTTP status code is returned.

Bitdefender Control Center exposes multiple APIs targeting distinct areas in the product. Each API exposes a set of methods related to a designated product area. The base URL for all APIs is the machine hostname, domain or IP where GravityZone is installed : https://YOUR-HOSTNAME/api/v1.0/jsonrpc/. To obtain the full URL of the API, add the API name to the base URL.

Currently, the following APIs are being exposed:

1. **Accounts**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/accounts.

2. **Network**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network.
3. **Packages**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/packages.
4. **Policies**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/policies.
5. **Reports**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/reports.

The HTTP requests containing JSON RPC 2.0 can be performed on each API URL in order to consume the exposed functionality.

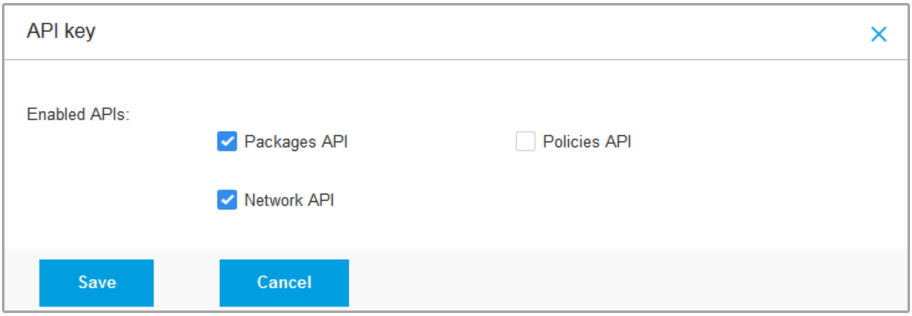
Note
Batch requests and notifications are not currently supported by Bitdefender Control Center.

1.3. API Keys

The API key is a unique key that is generated in **MyAccount** section of Bitdefender Control Center. Each API key allows the application to call methods exposed by one or several APIs. The allowed APIs are selected at the time the API key is generated.

To generate API keys:

1. Log in to https://YOUR-HOSTNAME/ using your administrative account. Your account must have the following rights: Manage Networks, Manage Users, Manage Company and Manage Reports.
2. Click your username in the upper-right corner of the console and choose **My Account**.
3. Go to the **API keys** section and click the ⊕ **Add** button at the upper side of the table.
4. Select the APIs that you want to use.

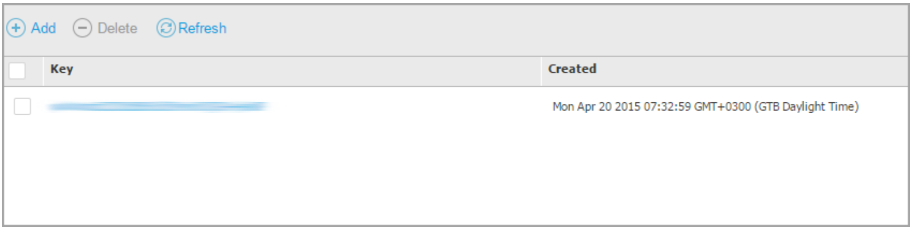


Enabled APIs:

☑ Packages API □ Policies API

☑ Network API

5. Click **Save**. An API key will be generated for the selected APIs.



□	Key	Created
□	[redacted]	Mon Apr 20 2015 07:32:59 GMT+0300 (GTB Daylight Time)

Important
By using the API keys, developers can access sensitive information such as packages and inventory. Please do not share or distribute your own generated API keys, in order to prevent the leaking of sensitive information!

1.4. Authentication

The API calls to Bitdefender Control Center are authenticated at HTTP protocol level using the `HTTP Basic Authentication` mechanism described here:

http://tools.ietf.org/html/rfc2617.

The client application is required to send the `Authorization` request header each time it performs a call to an API.

The `Authorization` header consists of the following elements:

1. The authorization method and a space as the prefix; in our case, this will always be equal to `Basic`.
2. A Base64 encoded string, generated from the combined `username:password` sequence.

   In our case, the API key is set as username, and password is set as an empty string.

   For example, if the API Key is equal to

   `N8KzwcqVUxAI1RoPi5jyFJPkPlkDl9vF`, the Base64 encoding should be performed on the following string:

   `N8KzwcqVUxAI1RoPi5jyFJPkPlkDl9vF:`. In this case, the content sent to the authorization header is

   `Basic TjhLendjcVZVeEFJMVJvUGk1anlGSlBrUGxrRGw5dkY6`.

1.5. Errors reporting

Bitdefender Control Center returns an error if the requested API method is unable to perform the desired task.

Here is an example of error response for a failing API call:

```
    {
            "id":"4d77e2d9-f760-4c8a-ba19-53728f868d98",
            "jsonrpc" : "2.0",
            "error" : {
                "code" : -32601,
                "message" : "Method not found",
                "data" : {
                    "details" : "The selected API is not available."
                }
            }
        }
```

The error code and error message are returned as specified in JSON-RPC 2.0 Specification:

Error	Code	Message
Parse error	-32700	Parse error
Invalid Request	-32600	Invalid Request
Method not found	-32601	Method not found
Invalid params	-32602	Invalid params
Server error	-32000	Server error

The full description of the error is placed in `data.details` member in the error message.

Also, the HTTP status code is set according to the type of errors:

HTTP status	Description
401 Unauthorized	is set if the authentication failed for the request (e.g. the API key is incorrect or missing)
403 Forbidden	is set if the request is not authorized to consume the desired functionality (e.g. the API is not enabled for the used API key)
405 Method Not Allowed	the HTTP method is other than POST
429 Too Many Requests	more than 10 requests per second have been issued from the same IP address

200 HTTP status code is returned for successful requests or for requests that have failed due to server errors (e.g. a required parameter is not passed).

2. REFERENCE

2.1. Accounts

The Accounts API includes several methods allowing the management of user accounts:

- `getAccountsList` : lists existing user accounts.
- `deleteAccount` : deletes a user account.
- `createAccount` : creates a user account.
- `updateAccount` : updates a user account.
- `configureNotificationsSettings` : configures the user notification settings.
- `getNotificationsSettings` : returns the notifications settings.

API url: https://YOUR-HOSTNAME/api/v1.0/jsonrpc/accounts

2.1.1. getAccountsList

This method lists the user accounts visible to the account which has generated the API key. It will return an empty list if there are no user accounts.

Note
When the accounts list is retrieved, the account which generated the API key **will be omitted**.

Parameters

Parameter	Type	Optional	Description
`page`	Number	Yes	The results page number. The default value is 1.
`perPage`	Number	Yes	The number of items displayed in a page. The upper limit is 30 items per page. Default value: 30 items per page.

Return value

This method returns an Object containing information regarding the user accounts. The returned object contains:

- `page` - the current page displayed
- `pagesCount` - the total number of available pages
- `perPage` - the total number of returned items per page
- `items` - the list of user accounts. Each entry in the list has the following fields:
  - `id`, the ID of the user account.
  - `userName`, the username of the user account.
  - `email`, the email of the user account.
  - `profile`, the profile information of the user account containing: `fullName`, `timezone` and `language`.
  - `role`, the role assigned for the user account. Possible values: 1 - Company Administrator, 2 - Network Administrator, 3 - Reporter, 5 - Custom.
  - `rights`, object containing the rights of the user account with true or false values whether the right is allowed for user or not.
- `total` - the total number of items

Example

Request :

```
{
  "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
  "jsonrpc": "2.0",
  "method": "getAccountsList",
  "params": {
          "perPage": 20,
          "page": 1
      }
 }
```

Response :

```
{
  "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
  "jsonrpc": "2.0",
  "result": {
      "total": 2,
      "page": 1,
      "perPage": 20,
      "pagesCount": 1,
      "items": [
          {
             "id": "585d3170aaed70b7048b4633",
             "userName": "client",
             "email": "client@bitdefender.com",
             "profile": {
                  "fullName": "Bitdefender User",
                  "language": "en_US",
                  "timezone": "Europe/Bucharest"
             },
            "role": 5,
            "rights": {
                  "companyManager": false,
                  "manageCompanies": false,
                  "manageNetworks": true,
                  "manageReports": true,
                  "manageUsers": true
              }
          },
          {
             "id": "585d3170aaed70b7048b4633",
             "userName": "client2",
             "email": "client2@bitdefender.com",
             "profile": {
                  "fullName": "Bitdefender User",
                  "language": "en_US",
                  "timezone": "Europe/Bucharest"
              },
              "role": 1,
              "rights": {
                  "companyManager": true,
                  "manageCompanies": false,
                  "manageNetworks": true,
                  "manageReports": true,
                  "manageUsers": true
              }
          }
       ]
    }
 }
```

2.1.2. deleteAccount

This method deletes a user account identified through the account ID.

Note
The account that was used to create the API key cannot be deleted by using the API.

Parameters

Parameter	Type	Optional	Description
`accountId`	String	No	The ID of the user account to be deleted.

Return value

This method does not return any value.

Example

Request :

```
{
  "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
  "jsonrpc": "2.0",
  "method": "deleteAccount",
  "params": {
       "accountId": "585d3810aaed70cc068b45f8"
      }
}
```

Response :

```
{
 "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
 "jsonrpc": "2.0",
 "result": null
}
```

2.1.3. createAccount

This method creates a user account with password.

Parameters

Parameter	Type	Optional	Description
`email`	String	No	The email address for the new account.
`userName`	String	No	The username for the account.
`profile`	Object	No	An object containing profile information: `fullName`, `timezone` and `language`. `timezone` and `language` are optional.
`password`	String	Yes	Password for the new account. If this value is omitted a password will be created and sent by email to the user. The password should be at least 6 characters in length and must contain at least one digit, one upper case, one lower case and one special character.
`role`	Number	Yes	The role of the new account. Default value is `1` - Company Administrator. These are the available roles: ● `1` - Company Administrator. ● `2` - Network Administrator. ● `3` - Reporter. ● `5` - Custom. For this role, rights must be specified.

Parameter	Type	Optional	Description
`rights`	Object	Yes	An object containing the rights of a user account. This object should be set only when `role` parameter has the value `5` - Custom. When set for other roles, the values will be ignored and replaced with the rights specific to that role. The available rights are: ● `manageCompanies` ● `manageNetworks` Setting this to true implies `manageReports` right to true ● `manageUsers` ● `manageReports` ● `companyManager` Each option has two possible values: true, where the user is granted the right, or false otherwise. Omitted values from the request are automatically set to `false`.
`targetIds`	Array	Yes	A list of IDs representing the targets to be managed by the user account.

Return value

This method returns a String: The ID of the created user account.

Example

Request :

```
   {
        "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
        "jsonrpc": "2.0",
        "method": "createAccount",
        "params": {
            "email": "client@bitdefender.com",
            "userName": "Client"
            "profile": {
                "fullName": "Bitdefender User",
                "language": "en_US",
                "timezone": "Europe/Bucharest"
            },
            "password": "P@s4w0rd",
            "role": 5,
            "rights": {
                "manageNetworks": true,
                "manageReports": true,
                "manageUsers": false
            },
            "targetIds": [
                "585d2dc9aaed70820e8b45b4",
                "585d2dd5aaed70b8048b45ca"
            ]
        }
   }
```

Response :

```
{
 "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
 "jsonrpc": "2.0",
 "result": "585d2dc9aaed70820abc45b4"
 }
```

2.1.4. updateAccount

This method updates a user account identified through the account ID.

Parameters

Parameter	Type	Optional	Description
`accountId`	String	No	The ID of the user account to be updated.
`email`	String	Yes	The new email address for the account.
`userName`	String	Yes	The new username for the user account.

Parameter	Type	Optional	Description
password	String	Yes	The new password for the user account. The password should at least 6 characters in length and must contain at least one digit, one upper case, one lower case and one special character.
profile	Object	No	An object containing profile information: fullName, timezone and language. timezone and language are optional.
role	Number	Yes	The new role of the user. These are the available roles: ● 1 - Company Administrator. ● 2 - Network Administrator. ● 3 - Reporter. ● 5 - Custom. For this role, rights must be specified.
rights	Object	Yes	An object containing the rights of a user account. This object should be set only when role parameter has the value 5 - Custom. When set for other roles, the values will be ignored and replaced with the rights specific to that role. The available rights are: ● manageCompanies ● manageNetworks Setting this to True implies manageReports right to true ● manageUsers ● manageReports ● companyManager Each option has two possible values: true, where the user is granted the right, or false otherwise. Omitted values from the request are automatically set to false.

Parameter	Type	Optional	Description
`targetIds`	Array	Yes	A list of IDs representing the targets to be managed by the user account.

Return value

This method returns a Boolean: True when the user account has been successfully updated.

Example

Request :

```
{
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
     "jsonrpc": "2.0",
     "method": "updateAccount",
     "params": {
         "accountId" : "585d3d3faaed70970e8b45ed",
         "email": "client@bitdefender.com",
         "profile": {
             "fullName": "Bitdefender User",
             "language": "en_US",
             "timezone": "Europe/Bucharest"
         },
         "password": "P@s4w0rd",
         "role": 5,
         "rights": {
             "manageNetworks": true,
             "manageReports": true,
             "manageUsers": false
         },
         "companyId": "58541613aaed7090058b4567",
         "targetIds": [
             "585d2dc9aaed70820e8b45b4",
             "585d2dd5aaed70b8048b45ca"
         ]
     }
}
```

Response :

```
  {
   "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
   "jsonrpc": "2.0",
   "result": true
   }
```

2.1.5. configureNotificationsSettings

This method configures the notification settings for a given user account.

Parameters

Parameter	Type	Optional	Description
`accountId`	String	Yes	The ID of the account for which the notification settings are configured. If no value is provided, the settings will be applied to the account which generated the API key.
`deleteAfter`	Number	Yes	The number of days after which generated notifications will be automatically deleted. Valid values are between 1 and 365. The default value is 30 days.
`emailAddresses`	Array	Yes	A list of additional email addresses to be used when sending notifications.
`includeDeviceName`	Boolean	Yes	This option specifies whether the device name will be included in the notification sent by email, when it is available, or not. The value should be `True` to include the device name respectively

Parameter	Type	Optional	Description
			`False` to not include it. The default value is `False`.
`includeDeviceFQDN`	Boolean	Yes	This option specifies whether the FQDN will be included in the notification sent by email, when it is available, or not. The value should be `True` to include the FQDN respectively `False` to not include it. The default value is `False`.
`notificationsSettings`	Array	Yes	A list of objects containing the notification settings to be configured. Only the specified notifications will be updated. Existing values are preserved for omitted settings. Each object should have the following structure: ● `type`, the notification type, ● `enabled`, `True` if the notification is enabled, `False` otherwise, ● `visibilitySettings`, an object containing the visibility settings. For more information, refer to Notifications Visibility Options, ● `configurationSettings`, notification specific configurations. This field depends on the notification `type`. For more information, refer to Relation Between

Parameter	Type	Optional	Description
			Notification Type and configurationSettings. .

Return value

This method returns a Boolean: True if the notifications settings have been successfully configured.

Example

Request :

```
   {
        "params": {
            "accountId": "55896b87b7894d0f367b23c8",
            "deleteAfter": 17,
            "includeDeviceName": true,
            "includeDeviceFQDN": true,
            "emailAddresses": ["example1@example.com"],
            "notificationsSettings":[
                {
                    "type" : 1,
                    "enabled" : true,
                    "visibilitySettings" : {
                        "sendPerEmail" : true,
                        "showInConsole" : true,
                        "useCustomEmailDistribution": false
                        "emails" : ["example2@example.com"],
                        "logToServer" : true
                    },
                    "configurationSettings" : {
                        "threshold" : 15,
                        "useThreshold" : true
                    }
                }
            ]
        },
        "jsonrpc": "2.0",
        "method": "configureNotificationsSettings",
        "id": "5399c9b5-0b46-45e4-81aa-889952433d68"
   }
```

Response :

```
   {
       "id":"5399c9b5-0b46-45e4-81aa-889952433d68",
       "jsonrpc":"2.0",
       "result": true
   }
```

2.1.6. getNotificationsSettings

This method returns the notifications settings.

Parameters

Parameter	Type	Optional	Description
`accountId`	String	Yes	The ID of the account for which the notifications settings are retrieved. If not provided, the method will retrieve the notifications settings for the account which has generated the API key.

Return value

This method returns an Object containing the current notifications settings:

- `deleteAfter` - the number of days after which generated notifications will be automatically deleted
- `includeDeviceName` - a boolean that informs whether the device name will be included in the notification sent by email or not
- `includeDeviceFQDN` - a boolean that informs whether the device FQDN will be included in the notification sent by email or not
- `emailAddresses` - the list of additional email addresses to be used when sending notifications

- `notificationsSettings` - the list containing the settings for all available notifications. Each entry in the list has the following fields:
  - `type`, the notification type,
  - `enabled`, `True` if the notification is enabled, `False` otherwise,
  - `visibilitySettings`, an object containing the configured visibility settings. For more information, refer to Notifications Visibility Options,
  - `configurationSettings`, notification specific configurations. For more information, refer to Relation Between Notification Type and configurationSettings.

Example

Request :

```
{
     "params": {
         "accountId": "55896b87b7894d0f367b23c8"
     },
     "jsonrpc": "2.0",
     "method": "getNotificationsSettings",
     "id": "5399c9b5-0b46-45e4-81aa-889952433d86"
}
```

Response :

```
{
    "id":"5399c9b5-0b46-45e4-81aa-889952433d86",
    "jsonrpc":"2.0",
    "result": {
         "deleteAfter": 21,
         "includeDeviceName": true,
         "includeDeviceFQDN": false,
         "emailAddresses": [
             "example1@example.com",
             "example2@example.com"
         ],
         "notificationsSettings":[
             {
                 "type" : 1,
                 "enabled" : true,
                 "visibilitySettings" : {
                     "sendPerEmail" : true,
                     "showInConsole" : true,
                     "useCustomEmailDistribution": false
                     "emails" : [],
                     "logToServer" : true
                 },
                 "configurationSettings" : {
                     "threshold" : 5,
                     "useThreshold" : true
                 }
             },
             {
                 "type" : 3,
                 "enabled" : false,
                 "visibilitySettings" : {
                     "sendPerEmail" : true,
                     "showInConsole" : true,
                     "useCustomEmailDistribution": false
                     "emails" : [],
                     "logToServer" : true
                 }
             },
             ...
         ]
     }
}
```

2.1.7. Objects

Notifications Visibility Options

You can use the `visibilitySettings` object to configure where notifications are visible. These are the available options:

Visibility option	Optional	Value
`showInConsole`	Yes	`True` to display this notification in Control Center, `False` otherwise. If

Visibility option	Optional	Value
		no value is specified it will be set to its previous value or `False` if a aprevious value was not set.
`sendPerEmail`	Yes	`True` to send this notification by email, `False` otherwise. If no value is specified it will be set to its previous value or `False` if a previous value was not set. This option will take effect only if a SMTP server is configured in the **Configuration** page of Bitdefender Control Center.
`useCustomEmailDistribution`	Yes	`True` to send email notification to a custom emailing list, `False` otherwise. The notification will be sent by email to the distribution list only. If this option is set to `True` the `sendPerEmail` parameter must be specified and set to `True`. If no value is specified it will be set to its previous value or `False` if a aprevious value was not set.
`emails`	Yes	A list of email addresses to receive the notification via email. When set, only these email addresses receive the notification. When `useCustomEmailDistribution` is set to `True`, this list must contain at least one valid email address.
`logToServer`	No	boolean, `True` to send this notification on the configured

Visibility option	Optional	Value
		syslog server, `False` otherwise. A syslog server must be configured in Control Center to receive this notification on the syslog server. This option is available only if a Syslog server is configured in the **Configuration** page of Bitdefender Control Center. If no value is specified it will be set to its previous value or `False` if a aprevious value was not set.

Note

- At least one visibility option from `showInConsole`, `sendPerEmail`, `logToServer` (when available) must be set to `True` when enabling the notification.
- The `sendPerEmail`, `useCustomEmailDistribution` and `emails` visibility options are not available for these notification types:
  - `6` - Internet Connection
  - `7` - SMTP Connection
  - `22` - Product Modules Event

Relation Between Notification Type and configurationSettings

Notification type	Available configurationSettings items with their type and possible values
1 - Malware Outbreak	• `useThreshold`, boolean, `True` to trigger this notification when the number of infected managed network objects exceeds a custom threshold, `False` otherwise

Notification type	Available configurationSettings items with their type and possible values
	• `threshold`, integer, the percentage of managed network objects infected by the same malware. Valid values are between 1 and 100
2 - License Expires	The `configurationSettings` parameter should not be set for this notification.
3 - License Usage Limit Has Been Reached	The `configurationSettings` parameter should not be set for this notification.
4 - License Limit Is About To Be Reached	The `configurationSettings` parameter should not be set for this notification.
5 - Update Available	• `showConsoleUpdate`, boolean, `True` to receive the notification for console updates, `False` otherwise • `showPackageUpdate`, boolean, `True` to receive the notification for package updates, `False` otherwise • `showProductUpdate`, boolean, `True` to receive the notification for product updates, `False` otherwise
6 - Internet Connection	The `configurationSettings` parameter should not be set for this notification.
7 - SMTP Connection	The `configurationSettings` parameter should not be set for this notification.
8 - Database Backup	• `onlyFailedEvents`, boolean, `True` to receive the notification for failed backup events only, `False` otherwise
9 - Exchange License Usage Limit Has Been Reached	The `configurationSettings` parameter should not be set for this notification.

Notification type	Available configurationSettings items with their type and possible values
10 - Invalid Exchange User Credentials	The `configurationSettings` parameter should not be set for this notification.
11 - Upgrade Status	The `configurationSettings` parameter should not be set for this notification.
12 - Exchange Malware Detected	The `configurationSettings` parameter should not be set for this notification.
13 - Authentication Audit	The `configurationSettings` parameter should not be set for this notification.
14 - Certificate Expires	The `configurationSettings` parameter should not be set for this notification.
15 - GravityZone Update	The `configurationSettings` parameter should not be set for this notification.
16 - Antimalware Event	The `configurationSettings` parameter should not be set for this notification.
17 - Antipshising Event	The `configurationSettings` parameter should not be set for this notification.
18 - Firewall Event	The `configurationSettings` parameter should not be set for this notification.
19 - ATC/IDS event	The `configurationSettings` parameter should not be set for this notification.
20 - User Control Event	The `configurationSettings` parameter should not be set for this notification.
21 - Data Protection Event	The `configurationSettings` parameter should not be set for this notification.
22 - Product Modules Event	The `configurationSettings` parameter should not be set for this notification.
23 - Security Server Status Event	● `notUpdated`, boolean, `True` to receive the notification when the Security Server is outdated, `False` otherwise

Notification type	Available configurationSettings items with their type and possible values
	• `reboot`, boolean, `True` to receive the notification when the Security Server needs a reboot, `False` otherwise • `stopped`, boolean, `True` to receive the notification when the Security Server was powered off, `False` otherwise
24 - Product Registration Event	The `configurationSettings` parameter should not be set for this notification.
25 - Overloaded Security Server Event	• `useThreshold`, boolean, `True` to receive the notification when the scan load exceeds a custom threshold, `False` otherwise • `threshold`, integer, the minimum scan load necessary to issue this notification. Valid values are between 1 and 100
26 - Task Status	• `statusThreshold`, integer, the task status which triggers this notification. Set to `2` for any status, `3` for failed tasks
27 - Outdated Update Server	The `configurationSettings` parameter should not be set for this notification.
28 - New Application In Application Inventory	The `configurationSettings` parameter should not be set for this notification.
29 - Blocked Application	• `fromProductionMode`, boolean, `True` to receive the notification for a blocked processes of an unauthorized application in Production Mode, `False` otherwise • `fromTestMode`, boolean, `True` to receive the notification for a blocked processes of an

Notification type	Available configurationSettings items with their type and possible values
	unauthorized application in Test Mode, False otherwise
30 - Detected Memory Violation	The configurationSettings parameter should not be set for this notification.
31 - Mobile Device Users Without EmailAddress	The configurationSettings parameter should not be set for this notification.

2.2. Network

The Network API allows managing the network structure through the following methods:

- getContainers : returns the network containers.
- createScanTask : returns true if the task was successfully created.
- getScanTasksList : returns the list of scan tasks.
- getEndpointsList : returns the list of endpoints.
- getManagedEndpointDetails : returns the details about a managed endpoint.
- createCustomGroup : creates a new group under an existing one or under **Computers and Groups**.
- deleteCustomGroup : deletes a custom group.
- moveCustomGroup : moves a custom group under another custom group.
- moveEndpoints : moves the specified list of endpoints to a custom group.
- deleteEndpoint : deletes a specified endpoint.

API url: https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/{service}

{service} is a placeholder that can hold specific values depending on the chosen API method. Please check the method documentation for the allowed services.

2.2.1. getContainers

This method returns network containers. It will return an empty list if the `parentId` is not a container or does not contain any other container within it.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"
- mobile, for "Mobile Devices"

Eg: The request URL for the `mobile` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/mobile

Parameters

Parameter	Type	Optional	Description
`parentId`	String	Yes	The ID of the container. If null, the top containers of the specified service type will be returned.
`viewType`	Number	Yes	The ID of the view type for the virtual environment inventory. The view type depends on the virtualization platform. In VMWare integrations, the available options are: ● 1 - Hosts and Clusters view (default) ● 2 - Virtual Machines view. In Citrix, XenServer integrations, the available options are: ● 3 - Server view (default) ● 4 - Folder view.

Return value

This method returns an Array containing a list of objects that represent the network containers. Each object has the following fields:

- `id` - the ID of the container
- `name` - the name of the container

Example

Request :

```
{
     "params": {
         "parentId": "559bd17ab1a43d241b7b23c6",
         "viewType": 4,
     },
     "jsonrpc": "2.0",
     "method": "getContainers",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": [
            {
                "id" : "5582c385b1a43deb7f7b23c6",
                "name" : "Xen Server"
            }
    ]
}
```

2.2.2. createScanTask

This method creates a new scan task.

Note
Please note that the managed endpoints from `virtualmachines` service are also displayed in `computers` service under **Custom Group** To avoid launching duplicate scan tasks we recommend you to use the endpoints from the `computers` service.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`targetIds`	Array	No	A list containing the IDs of endpoints or containers to scan.
`type`	Number	No	The type of scan. Available options are: 1 - quick scan; 2 - full scan; 3 - memory scan
`name`	String	Yes	The name of the task. If the parameter is not passed, the name will be automatically generated.

Return value

This method returns a Boolean: True when the task was successfully created

Example

Request :

```
{
     "params": {
         "targetIds": ["559bd17ab1a43d241b7b23c6",
                          "559bd17ab1a43d241b7b23c7"],
         "type": 1,
         "name": "my scan"
     },
     "jsonrpc": "2.0",
     "method": "createScanTask",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
     "jsonrpc":"2.0",
     "result": True
}
```

2.2.3. getScanTasksList

This method returns the list of scan tasks.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`name`	String	Yes	The name of the task. Filters the list of tasks by task name. Use the asterisk symbol (*) in front of the keyword to search its appearance anywhere in the name. If omitted, only results where the name starts with the keyword will be returned.
`status`	Number	Yes	The status of the task. Available options are: 1 - Pending; 2 - In progress; 3 - Finished.

Parameter	Type	Optional	Description
`page`	Number	Yes	The results page number. Default page number is 1.
`perPage`	Number	Yes	Number of items per page to be returned. The upper limit is 30 items per page. Default value: 30 items per page.

Return value

This method returns an Object containing information about the tasks. The returned object contains:

- `page` - the results page number
- `pagesCount` - the total number of available pages
- `perPage` - the total number of returned items per page
- `total` - the total number of items
- `items` - the list of tasks. Each entry in the list has the following fields:
  - `id`, the ID of the task,
  - `name`, the name of the task,
  - `status`, the status of the task (as defined above),
  - `startDate`, the start date of the task

Example

Request :

```
{
     "params": {
         "status": 1,
         "page": 2,
         "perPage": 5
     },
     "jsonrpc": "2.0",
     "method": "getScanTasksList",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": {
        page: 2,
        pagesCount: 11,
        perPage: 5,
        total: 54
        items[
            {
                "id" : "21a295eeb1a43d8b497b23b7",
                "name" : "task 1",
                "status": 1,
                "startDate": '2015-08-21T23:48:16'
            },
            {
                "id" : "21a295eeb1a43d8b497b23b8",
                "name" : "task 2",
                "status": 1,
                "startDate": '2015-08-21T10:21:15'
            },
        ]
    }
}
```

2.2.4. getEndpointsList

This method returns the list of the endpoints.

To find the `parentId`, you must do several recursive calls to `getContainers` until the container with the endpoints is reached. The container ID from the response of `getContainers` should be used as parentId in this call. The same `viewType` used in getContainers should be used in this call.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`parentId`	String	Yes	The ID of the container for which the endpoints list will be returned. If null, the endpoints within the root custom group of the specified service are returned.
`isManaged`	Boolean	No	The flag to list managed or unmanaged endpoints. By default, the parameter is not set and the method returns all managed and unmanaged endpoints. If set on `True`, the method returns only managed endpoints.
`viewType`	Number	Yes	The ID of the view type for the virtual environment inventory. The view type depends on the virtualization platform. In VMWare integrations, the available options are: ● 1 - Hosts and Clusters view (default) ● 2 - Virtual Machines view. In Citrix, XenServer integrations, the available options are: ● 3 - Server view (default) ● 4 - Folder view.
`page`	Number	Yes	The results page number. Default page number is 1.
`perPage`	Number	Yes	The number of items displayed in a page. The upper limit is 30 items per page. Default value: 30 items per page.

Return value

This method returns an Object containing information about the endpoints. The returned object contains:

- `page` - the current page displayed
- `pagesCount` - the total number of available pages
- `perPage` - the total number of returned items per page
- `total` - the total number of items
- `items` - the list of endpoints. Each entry in the list has the following fields: `id`, string, the ID of the endpoint, `name`, string, the name of the endpoint, `machineType`, int, the type of the machine the type of the machine: (1 - computer, 2 - virtual machine, 0 - Other)

Example

Request :

```
{
     "params": {
         "parentId": "23b19c39b1a43d89367b32ce",
         "page": 2,
         "perPage": 5
     },
     "jsonrpc": "2.0",
     "method": "getEndpointsList",
     "id": "301f7b05-ec02-481b-9ed6-c07b97de2b7b"
}
```

Response :

```
{
    "id":"103d7b05-ec02-481b-9ed6-c07b97de2b7a",
    "jsonrpc":"2.0",
    "result": {
         page: 2,
         pagesCount: 11,
         perPage: 5,
         total: 54
         items[
             {
                 "id" : "21a295eeb1a43d8b497b23b7",
                 "name" : "Endpoint 1",
                 "machineType": 1,
             },
             {
                 "id" : "23a295d8b1a43d7c4a7b23c9",
                 "name" : "Endpoint 2",
                 "machineType": 2,
             }
         ]
    }
}
```

2.2.5. getManagedEndpointDetails

This method returns detailed information, such as: the identification details for endpoint and security agent, the status of installed protection modules, and scanning reports and logs about a managed endpoint.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`endpointId`	String	No	The ID of the endpoint for which the details will be returned

Return value

This method returns an Object containing the details of the specified endpoint:

- `id` - the ID of managed endpoint
- `name` - the name of the endpoint
- `operatingSystem` - the Operating System of the endpoint
- `state` - the power state of the machine: 1 - online, 2 - offline, 3 - suspended; 0 - unknown.
- `ip` - the IP of the endpoint
- `lastSeen` - the date of the last synchronization with Control Center
- `machineType` - the type of the machine: 1 - computer, 2 - virtual machine, 0 - Other
- `agent` - an object with the agent information from the endpoint.

  Object description:

  - `engineVersion`, string, the version of the engine
  - `primaryEngine`, integer, can be 1 (for Central Scanning (Security Server)), 2 (for Hybrid Scanning (Light Engines)) or 3 (for Local Scanning (Full Engines)); 0 Unknown
  - `fallbackEngine`, integer, can be 2 (for Hybrid Scanning (Light Engines)) or 3 (for Local Scanning (Full Engines)); 0 Unknown
  - `lastUpdate`, date, the last update of the signatures
  - `licensed`, integer, the status of the license: 0 - pending authentication, 1 - active license, 2 - expired license, 6 - there is no license or not applicable
  - `productOutdated`, boolean, specifies if the product is outdated
  - `productUpdateDisabled`, boolean, specifies if the updates for the product is disabled
  - `productVersion`, string, the version of the product
  - `signatureOutdated`, boolean, specifies if the signatures of the endpoint are outdated

  - `signatureUpdateDisabled`, boolean, specifies if the update for the signatures of the endpoint is disabled
  - `type`, integer, the type of the endpoint. It can be: 1 - Endpoint Security, 2 - Bitdefender Tools, 3 - Bitdefender Endpoint Security.
- `group` - object, information about the group of the endpoint. It contains `id`, string, the id of the group and `name`, string, the name of the group
- `malwareStatus` - object, information about the malwareStatus on the endpoint. It contains `detection`, boolean, if there is any malware detection on the endpoint, and `infected`, boolean, if the endpoint is infected
- `policy` - object, information about the active policy on the endpoint. It contains: `id`, string, the ID of the active policy, `name`, string, the name of the policy, `applied`, boolean, true if the policy is applied
- `hypervisorMemoryIntrospection` - object, information about hypervisor memory introspection. This object appears only if the endpoint is managed by HVI.

  Object description:

  - `status`, boolean, specifies if hypervisor memory introspection is enabled for the endpoint.
  - `activeModules`, object, information about the modules for Hypervisor memory introspection. It contains `userMode`, boolean, indicating whether User Memory introspection is active and `kernelMode`, boolean, indicating whether Kernel Memory introspection is active for the endpoint.
  - `securityServer`, object, information about the security server which protects the endpoint. It contains `name`, string, the name of the security server, `ip`, string, the IP of the security server and `Label`, string, the label associated with the server
  - `isLicensed`, boolean, specifies if the endpoint is licensed for Hypervisor memory introspection
- `modules` - object, the modules and their status; Possible keys are: advancedThreatControl, antimalware, contentControl, deviceControl, firewall,

powerUser. The values are true, if the module is enabled or false, if the module is not enabled.

Example

Request :

```
{
     "params": {
         "endpointId" : "54a28b41b1a43d89367b23fd"
     },
     "jsonrpc": "2.0",
     "method": "getManagedEndpointDetails",
     "id": "301f7b05-ec02-481b-9ed6-c07b97de2b7b"
}
```

Response :

```
{
    "id":"0df7568c-59c1-48e0-a31b-18d83e6d9810",
    "jsonrpc":"2.0",
    "result": {
        'id': '54a28b41b1a43d89367b23fd',
        'name': 'WIN-TGQDU499RS4',
        'operatingSystem': 'Windows Server 2008 R2 Datacenter',
        'state': 1,
        'ip': '10.10.24.154',
        'lastSeen': '2015-06-22T13:46:59',
        'machineType': 1,
        'agent': {
             'engineVersion': '7.61184',
             'primaryEngine': 1,
             'fallbackEngine': 2,
             'lastUpdate': '2015-06-22T13:40:06',
             'licensed': 1,
             'productOutdated': False,
             'productUpdateDisabled': False,
             'productVersion': '6.2.3.569',
             'signatureOutdated': False,
             'signatureUpdateDisabled': False,
             'type': 3
         },
        'group': {
             'id': '5575a235d2172c65038b456d',
             'name': 'Custom Groups'
         },
        'malwareStatus': {
             'detection': False,
             'infected': False
         },
        'modules': {
             'advancedThreatControl': False,
             'antimalware': True,
             'contentControl': False,
             'deviceControl': False,
             'firewall': False,
             'powerUser': False
         },
         'hypervisorMemoryIntrospection': {
             'status': 'enabled',
             'activeModules': {
                 'userMode': true,
                 'kernelMode': false
                 },
             'securityServer': {
                 'name': 'Security Server',
                 'ip': '192.168.0.100',
                 'label': 'N/A'
                 },
             'isLicensed': true
         },
          'policy': {
                 'id': '5121da426803fa2d0e000017',
                 'applied': True,
                 'name': 'Default policy'
         }
     }
}
```

2.2.6. createCustomGroup

This method creates a new custom group.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`groupName`	String	No	The name for the new group
`parentId`	String	Yes	The ID of the parent group. If `parentId` is null, the new group is created under **Custom Groups**.

Return value

This method returns a String: the ID of the new created group.

Example

Request :

```
{
      "params": {
           "groupName": "myGroup",
           "parentId": "5582c0acb1a43d9f7f7b23c6"
      },
      "jsonrpc": "2.0",
      "method": "createCustomGroup",
      "id": "9600512e-4e89-438a-915d-1340c654ae34"
}
```

Response :

```
{
    "id": "9600512e-4e89-438a-915d-1340c654ae34",
    "jsonrpc":"2.0",
    "result": "5582c210b1a43d967f7b23c6"
}
```

2.2.7. deleteCustomGroup

This method deletes a custom group.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`groupId`	String	No	The ID of the custom group to be deleted
`force`	Boolean	Yes	Force delete when group is not empty. By default, the parameter is set to `False`.

Return value

This method does not return any value.

Example

Request :

```
{
        "params": {
            "groupId": "559bd17ab1a43d241b7b23c6",
            "force": true
        },
        "jsonrpc": "2.0",
        "method": "deleteCustomGroup",
        "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
       "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
       "jsonrpc":"2.0",
       "result": null
}
```

2.2.8. moveCustomGroup

This method moves a custom group to another custom group.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`groupId`	String	No	The ID of the custom group to be moved

Parameter	Type	Optional	Description
`parentId`	String	No	The ID of the destination custom group

Return value

This method does not return any value.

Example

Request :

```
{
     "params": {
         "groupdId": "559bd17ab1a43d241b7b23c6",
         "parentId": "559bd17ab1a85d241b7b23c6"
     },
     "jsonrpc": "2.0",
     "method": "moveCustomGroup",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": null
}
```

2.2.9. moveEndpoints

This method moves a list of endpoints to a custom group.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"

- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`endpointIds`	Array	No	The list of endpoints IDs
`groupId`	String	No	The ID of the destination group

Return value

This method does not return any value.

Example

Request :

```
{
     "params": {
         "endpointIds" : [
                 "559bd152b1a43d291b7b23d8",
                 "559bd152b1a43d291b7b2430"
         ],
         "groupdId": "559bd17ab1a43d241b7b23c6"
     },
     "jsonrpc": "2.0",
     "method": "moveEndpoints",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": null
}
```

2.2.10. deleteEndpoint

This method deletes an endpoint.

Note
Deleting an endpoint under `Custom Groups` moves it to the `Deleted` group. For managed endpoints, an `Uninstall` task is automatically generated. To permanently remove an endpoint, call the method twice using the same ID.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/network/virtualmachines

Parameters

Parameter	Type	Optional	Description
`endpointId`	String	No	The ID of the endpoint

Return value

This method does not return any value.

Example

Request :

```
{
      "params": {
           "endpointId" : "559bd152b1a43d291b7b23d8"
      },
      "jsonrpc": "2.0",
      "method": "deleteEndpoint",
      "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
   {
       "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
       "jsonrpc":"2.0",
       "result": null
   }
```

2.3. Packages

The Packages API contains the following methods allowing the management of installation packages:

- `getPackagesList` : returns the list of available packages.

API url: https://YOUR_HOSTNAME/api/v1.0/jsonrpc/packages

2.3.1. getPackagesList

Returns the list of available packages.

Parameters

Parameter	Type	Optional	Description
`page`	Number	Yes	The results page number. Default page number is 1.
`perPage`	Number	Yes	Number of items per page to be returned. The upper limit is 30 items per page. Default value: 30 items per page.

Return value

This method returns an Object containing information about the packages. The returned object contains:

- `page` - the current page displayed
- `pagesCount` - the total number of available pages
- `perPage` - the total number of returned items per page
- `total` - the total number of items
- `items` - the list of packages. Each entry in the list has the following fields: `id`, the ID of the package; `name`, the name of the package; `type`, the type of the package. It can be 3 for SVA, 4 for Bitdefender Endpoint Security.

Example

Request :

```
{
     "params": {
         "page": 1,
         "perPage": 5
     },
     "jsonrpc": "2.0",
     "method": "getPackagesList",
     "id": "696e1024-f94b-496a-9394-bee58b73c51f"
}
```

Response :

```
{
    "id":"103d7b05-ec02-481b-9ed6-c07b97de2b7a",
    "jsonrpc":"2.0",
    "result": {
         "page": 1,
         "pagesCount": 1,
         "perPage": 5,
         "total": 1,
         "items": [
             {
                 "id" : "55b8c1bfb1a43dd71071071b",
                 "name" : "Package Test",
                 "type": 3
             }
         ]
    }
}
```

2.4. Policies

The Policies API includes several methods allowing the management of security policies:

- `getPoliciesList` : retrieves the list of available policies.
- `getPolicyDetails` : retrieves the settings of a security policy.

API url: https://YOUR_HOSTNAME/api/v1.0/jsonrpc/policies/{service}

`{service}` is a placeholder that can hold specific values depending on the chosen API method. Please check the method documentation for the allowed services.

Note
Please note that a security policy can be applied on both computers and virtual machines. Therefore, the methods exposed using this API require only the `computers` service.

2.4.1. getPoliciesList

This method retrieves the list of available policies.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"

Eg: The request URL for the `computers` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/policies/computers

Parameters

Parameter	Type	Optional	Description
`page`	Number	Yes	The results page. The default value is 1.
`perPage`	Number	Yes	How many items per page should be returned. The default value is 30 items.

Return value

This method returns an Array containing policy objects identifying the policies available to the specified company. Each entry in the array has the following structure:

- `page` - int, the current displayed page
- `pagesCount` - int, the total number of available pages
- `perPage` - int, the total number of returned items per page
- `total` - int, the total number of items
- `items` - array, the list of policies. Each entry in the list has the following fields: `id`, string, the ID of the policy, `name`, string, the name of the policy

Example

Request :

```
{
     "params": {
         "page": 1,
         "perPage": 2
     },
     "jsonrpc": "2.0",
     "method": "getPoliciesList",
     "id": "5399c9b5-0b46-45e4-81aa-889952433d86"
}
```

Response :

```
{
    "id":"5399c9b5-0b46-45e4-81aa-889952433d86",
    "jsonrpc":"2.0",
    "result": {
         page: 1,
         pagesCount: 2,
         perPage: 2,
         total: 4
         items[
             {
                 "id" : "21a295eeb1a43d8b497b23b7",
                 "name" : "Policy 1"
             },
             {
                 "id" : "23a295d8b1a43d7c4a7b23c9",
                 "name" : "Policy 2"
             }
         ]
     }
}
```

2.4.2. getPolicyDetails

This method retrieves all the information related to a security policy.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"

Eg: The request URL for the `computers` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/policies/computers

Parameters

Parameter	Type	Optional	Description
`policyId`	String	No	The ID of the policy to be queried.

Return value

This method returns an Object containing the details of the queried policy:

- `id` - the ID of the queried policy
- `name` - the name of the queried policy
- `createdBy` - the username of the user who created the policy
- `createDate` - the date when the policy was created
- `lastModifyDate` - the date when the policy was last modified
- `settings` - the settings of the policy

Example

Request :

```
{
     "params": {
         "policyId" : "55828d66b1a43de92c712345"
     },
     "jsonrpc": "2.0",
     "method": "getPolicyDetails",
     "id": "98409cc1-93cc-415a-9f77-1d4f681000b3"
}
```

Response :

```
{
    "id": "47519d2d-92e0-4a1f-b06d-aa458e80f610",
    "jsonrpc":"2.0",
    "result": {
         "id": "5583c480b1a43ddc09712345",
         "name": "Test",
         "createdBy": "user@bitdefender.com",
         "createDate": "2015-06-19T10:27:59",
         "lastModifyDate": "2015-06-19T10:27:59",
         "settings": {
             ...
```

```
            }
        }
}
```

2.5. Reports

The Reports API includes several methods allowing the reports management:

- `createReport` : creates a new instant or scheduled report and returns the ID of the newly-created report.
- `getReportsList` : returns the list of reports.
- `getDownloadLinks` : returns the download links for a report.
- `deleteReport` : deletes the specified report and returns true on success or an error status code and error message on fail.

API url: https://YOUR-HOSTNAME/api/v1.0/jsonrpc/reports

2.5.1. createReport

This method creates a new instant or scheduled report, based on the parameters received, and returns the ID of the new created report.

The instant report is created and runs one-time-only at the API call.

The scheduled report is created at a later time and runs periodically, based on a predefined schedule.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/reports/virtualmachines

Parameters

Parameter	Type	Optional	Description
`name`	String	No	The name of the report.
`type`	Number	Yes	The type of report. The acceptable values are: ● 1 - Antiphishing Activity ● 2 - Blocked Applications ● 3 - Blocked Websites ● 5 - Data Protection ● 6 - Device Control Activity ● 7 - Endpoint Modules Status ● 8 - Endpoint Protection Status ● 9 - Firewall Activity ● 11 - Malware Activity ● 12 - Malware Status ● 14 - Network Status ● 15 - On demand scanning ● 16 - Policy Compliance ● 17 - Security Audit ● 18 - Security Server Status ● 19 - Top 10 Detected Malware ● 21 - Top 10 Infected Endpoints ● 22 - Update Status ● 25 - Virtual Machine Network Status ● 26 - HVI Activity
`targetIds`	Array	No	A list with the IDs of the targets for which to create the report. The target ID can be any

Parameter	Type	Optional	Description
			of the following: groups, containers or endpoints.
`scheduledInfo`	Object	Yes	The object that defines the schedule to run the report. If the parameter is omitted, an instant report is generated. For more information, please check the details of the scheduledInfo object.
`options`	Object	Yes	The object that defines the options for creating the report. For these reports, the `options` object should not be set: ● Endpoint Modules Status ● Policy Compliance ● Security Server Status For more information, please check the details of the options object.
`emailsList`	Array	Yes	A list of emails where to deliver the report. `emailsList` should not be set for an instant report.

Objects

scheduledInfo

This object is used by the `createReport` call and it defines the schedule based on which the report will run.

The object contains a variable number of members, depending on the occurrence of the report:

Name	Type	Description
`occurrence`	integer	The member is mandatory. Possible values: – 1 - for an instant report

Name	Type	Description
		– 2 - for hourly report – 3 - for daily report – 4 - for weekly report – 5 - for monthly report – 6 - for yearly report
`interval`	integer	The member should be set only if `occurrence` has the value 2. Possible values: – Any integer between 1 and 24, representing the interval (in hours) at which the report will run.
`startHour`	integer	The member should be set only if `occurrence` has the value 3, 4 or 5. Possible values: – Any integer between 0 and 23.
`startMinute`	integer	The member should be set only if `occurrence` has the value 3, 4 or 5. Possible values: – Any integer between 0 and 59.
`days`	array	The member should be set only if `occurrence` has the value 4. Possible values of the array elements: – Integers between 0 and 6, representing the days of the week, from 0 - Sunday to 6 - Saturday.
`day`	integer	The member should be set only if `occurrence` has the value 5 or 6. Possible values:

Name	Type	Description
		– An integer between 1 and 31, representing the day of the month.
`month`	integer	The member should be set only if `occurrence` has the value 6. Possible values: – An integer between 1 and 12, representing the month of the year.

options

This object is used by the `createReport` call and contains a variable number of members, depending on the report type:

- **Antiphishing Activity**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints – 1 - Only endpoints with blocked websites

- **Blocked Applications**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory.

Name	Type	Description
		This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence

- **Blocked Websites**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints – 1 - Only endpoints with blocked websites

- **Data Protection**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints

Name	Type	Description
		– 1 - Only managed computers with blocked threats
`blockedEmails`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True` – `False`
`blockedWebsites`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True` – `False`

- **Device Control Activity**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence

- **Endpoint Protection Status**

  The object must contain these members:

Name	Type	Description
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints

Name	Type	Description
		– 1 - Only endpoints filtered by the members described hereinafter.
`antivirusOn`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True`, to include in the report endpoints with antimalware protection enabled. – `False`, to exclude from the report endpoints with antimalware protection enabled.
`antivirusOff`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True`, to include in the report endpoints with antimalware protection disabled. – `False`, to exclude from the report endpoints with antimalware protection disabled.
`updated`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True`, to include in the report updated endpoints. – `False`, to exclude from the report updated endpoints.
`disabled`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True`, to include in the report endpoints with update disabled. – `False`, to exclude from the report endpoints with update disabled.

Name	Type	Description
`outdated`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True`, to include in the report outdated endpoints. – `False`, to exclude from the report outdated endpoints.
`online`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True`, to include in the report online endpoints. – `False`, to exclude from the report online endpoints.
`offline`	boolean	The member should be set only if `filterType` has the value 1. Possible values: – `True`, to include in the report offline endpoints. – `False`, to exclude from the report offline endpoints.

- **Firewall Activity**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence
`filterType`	integer	The member is mandatory.

Name	Type	Description
		Possible values: – 0 - All endpoints – 1 - Only endpoints with the following blocked threats: traffic attempts and port scans.
`trafficAttempts`	boolean	This member should be set only if filterType has the value 1. Possible values: – `True`, to include in the report endpoints with blocked traffic attepts. – `False`, to exclude from the report endpoints with blocked traffic attepts.
`portScans`	boolean	This member should be set only if filterType has the value 1. Possible values: – `True`, to include in the report endpoints with blocked port scans. – `False`, to exclude from the report endpoints with blocked port scans.

- **Malware Activity**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints

Name	Type	Description
		– 1 - Only endpoints with unresolved malware

- **Malware Status**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints – 1 - Only endpoints still infected

- **Network Status**

  The object must contain these members:

Name	Type	Description
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints – 1 - Only endpoints with issues – 2 - Only endpoints with unknown status

- **On demand scanning**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence

- **Security Audit**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence

- **Top 10 Detected Malware**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence

- **Top 10 Infected Endpoints**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence

- **Update Status**

  The object must contain these members:

Name	Type	Description
`updated`	boolean	Possible values: – `True`, to include in the report updated endpoints. – `False`, to exclude from the report updated endpoints.
`disabled`	boolean	Possible values: – `True`, to include in the report endpoints with update disabled. – `False`, to exclude from the report endpoints with update disabled.
`outdated`	boolean	Possible values: – `True`, to include in the report outdated endpoints. – `False`, to exclude from the report outdated endpoints.
`pendingRestart`	boolean	Possible values: – `True`, to include in the report endpoints that need to be restarted. – `False`, to exclude from the report endpoints that need to be restarted.

- **VM Network Protection Status**

  The object must contain these members:

Name	Type	Description
`filterType`	integer	The member is mandatory.

Name	Type	Description
		Possible values: – 0 - All endpoints – 1 - Only protected endpoints

- **HVI Activity**

  The object must contain these members:

Name	Type	Description
`reportingInterval`	integer	The member is mandatory. This value depends on the report `occurrence`. For more information, refer to Relation between reporting interval and reccurence

Important

The object should not be set for these reports:

- **Endpoint Modules Status**
- **Policy Compliance**
- **Security Server Status**

Relation between reporting interval and reccurence

`occurrence`	`reportingInterval`
2 - Hourly report	Possible values: – 0 - Today
3 - Daily report	Possible values: – 0 - Today – 1 - Last day – 2 - This Week

occurrence	reportingInterval
4 - Weekly report	Possible values: – 0 - Today – 1 - Last day – 2 - This Week – 3 - Last Week – 4 - This Month
5 - Monthly report	Possible values: – 0 - Today – 1 - Last day – 2 - This week – 3 - Last week – 4 - This month – 5 - Last month – 6 - Last 2 months – 7 - Last 3 months – 8 - This year
6 - Yearly report	Possible values: – 8 - This year – 9 - Last year

Return value

This method returns a String: the ID of the created report.

Example

Request :

```
{
     "params": {
         "name": "My Report hourly",
         "type": 1,
         "targetIds": ["559bd17ab1a43d241b7b23c6",
                      "559bd17ab1a43d241b7b23c7"],
         "scheduledInfo": {
              "occurrence": 2,
              "interval": 4
          },
          "emailList": ["user@company.com",
                        "user2@company.com"]
     },
     "jsonrpc": "2.0",
     "method": "createReport",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Request :

```
{
     "params": {
         "name": "My Report daily",
         "type": 8,
         "targetIds": ["559bd17ab1a43d241b7b23c6",
                      "559bd17ab1a43d241b7b23c7"],
         "scheduledInfo": {
              "occurrence": 3,
              "startHour": 10,
              "startMinute": 30
          },
          "options": {
              "filterType": 1,
              "antivirusOn": true,
              "antivirusOff": false,
              "updated": true,
              "disabled": false,
              "outdated": false,
              "online": false,
              "offline": true
```

```
            }
        },
        "jsonrpc": "2.0",
        "method": "createReport",
        "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
       "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
       "jsonrpc":"2.0",
       "result": "563c78e2b1a43d4043d60413"
}
```

2.5.2. getReportsList

This method returns the list of scheduled reports, according to the parameters received.

Services

This method requires the {service} to be placed in the API url. The allowed services are:

- computers, for "Computers and Virtual Machines"
- virtualmachines, for "Virtual Machines"

Eg: The request URL for the `virtual machines` service is:

https://YOUR-HOSTNAME/api/v1.0/jsonrpc/reports/virtualmachines

Parameters

Parameter	Type	Optional	Description
`name`	String	Yes	The name of the report.
`type`	Number	No	The report type. The available types are: ● 1 - Antiphishing Activity

Parameter	Type	Optional	Description
			• 2 - Blocked Applications • 3 - Blocked Websites • 5 - Data Protection • 6 - Device Control Activity • 7 - Endpoint Modules Status • 8 - Endpoint Protection Status • 9 - Firewall Activity • 11 - Malware Activity • 12 - Malware Status • 14 - Network Status • 15 - On demand scanning • 16 - Policy Compliance • 17 - Security Audit • 18 - Security Server Status • 19 - Top 10 Detected Malware • 21 - Top 10 Infected Endpoints • 22 - Update Status • 25 - Virtual Machine Network Status • 26 - HVI Activity
`page`	Number	Yes	The results page number. Default page number is 1.
`perPage`	Number	Yes	The number of items displayed in a page. The upper limit is 30 items per page. Default value: 30 items per page.

Return value

This method returns an Object containing information about the reports. The returned object contains:

- page - the current page displayed
- pagesCount - the total number of available pages
- perPage - the total number of returned items per page
- items - the list of reports. Each entry in the list has the following fields:
  - ID, the ID of the report
  - name, the name of the report
  - type, the report type, as described in the Parameters table
  - occurrence, the time interval when the report runs. The occurrence can be: 2 - hourly, 3 - daily, 4 - weekly or 5 - monthly. Please mind that value 1 (instant report) is excluded from the valid options.
- total - the total number of items

Example

Request :

```
{
     "params": {
         "type": 2,
         "page": 2,
         "perPage": 4
     },
     "jsonrpc": "2.0",
     "method": "getReportsList",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": {
         "page": 2,
         "pagesCount": 11,
         "perPage": 5,
         "total": 54
         "items": [
             {
                 'id': '5638cdceb1a43d46137b23c6',
                 'name': 'My report 1',
                 'occurrence': 2,
                 'type': 2
             },
             {
                 'id': '5638d7f8b1a43d49137b23c9',
                 'name': 'My report 2',
                 'occurrence': 4,
                 'type': 2
             },
             {
                 'id': u'563b271bb1a43d21077b23c8',
                 'name': 'My report 3',
                 'occurrence': 4,
                 'type': 2
             },
             {
                 'id': '563a289eb1a43d2f617b23c6',
                 'name': 'My report 4',
                 'occurrence': 2,
                 'type': 2
             }
         ]
    }
}
```

2.5.3. getDownloadLinks

This method returns an Object with information regarding the report availability for download and the corresponding download links.

The instant report is created one time only and available for download for less than 24 hours.

Scheduled reports are generated periodically and all report instances are saved in the GravityZone database.

Parameters

Parameter	Type	Optional	Description
`reportId`	String	No	The report ID

Return value

This method returns an Object containing information for downloading the report. The returned object contains:

- `readyForDownload` - boolean, `True` if the report is ready to be downloaded or `False` otherwise
- `lastInstanceUrl` - string, The URL for downloading the last instance of an instant or scheduled report. It will be present in the response only if `readyForDownload` is `True`. The downloaded result is an archive with two files: a CSV and a PDF. Both files refer to the same last instance of the report.

  Note
  To access this URL, the HTTP basic authentication header (username:password pair) needs to be sent, where the username it is your API key and the password is a an empty string. For more information, refer to 1.3 Authentication section for details.

- `allInstancesUrl` - string, The URL downloads an archive with all generated instances of the scheduled report. The field will be present in the response only if `readyForDownload` is `True` and the report is a scheduled one. The downloaded result is an archive with a pair of files for each instance of the report: a CSV and a PDF file. Both files refer to the same instance of the report.

  Note
  To access this URL, the HTTP basic authentication header (username:password pair) needs to be sent, where the username it is your API key and the password is a an empty string. For more information, refer to 1.3 Authentication section for details.

Example

Request :

```
{
     "params": {
         "reportId": "5638d7f8b1a43d49137b23c9"
     },
     "jsonrpc": "2.0",
     "method": "getDownloadLinks",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87g"
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": {
        "readyForDownload": True,
        "allInstancesUrl":
           "https://YOUR-HOSTNAME/api/
            v1.0/http/downloadReportZip?reportId=
            5645cba6f12a9a8c5e8b4748&
            allInstances=1&serviceType=1",
        "lastInstanceUrl":
           "https://YOUR-HOSTNAME/api/
            v1.0/http/downloadReportZip?reportId=
            5645cba6f12a9a8c5e8b4748&
            allInstances=0&serviceType=1",
    }
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": {
        "readyForDownload": False
    }
}
```

Request :

```
Eg: Download the report using curl:

curl -f0 -u "YOUR_API_KEY:" \
https://YOUR-HOSTNAME/api/v1.0/http/\
downloadReportZip?reportId=5645cba6f12a9a8c5e8b4748&\
allInstances=0&serviceType=1 > lastReportInstances.zip

Equivalent with:

curl -f0 -H "Authorization: Basic API_KEY_ENCODED_BASE64" \
https://YOUR-HOSTNAME/api/v1.0/http/\
downloadReportZip?reportId=5645cba6f12a9a8c5e8b4748&\
allInstances=0&serviceType=1 > lastReportInstances.zip

Where API_KEY_ENCODED_BASE64 is your API key encoded
using base64.
```

2.5.4. deleteReport

The method deletes a report by its ID.

Parameters

Parameter	Type	Optional	Description
`reportId`	String	No	The report ID

Return value

This method returns a Boolean: True when the report was successfully deleted.

Example

Request :

```
{
     "params": {
         "reportId": "5638d7f8b1a43d49137b23c9"
     },
     "jsonrpc": "2.0",
     "method": "deleteReport",
     "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87g"
}
```

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": True
}
```

3. API USAGE EXAMPLES

The following API usage examples make use of the following generated API key: "UjlMS+0m1l9IUZjpjWyJG8gbnv2Mta4T".

3.1. C# Example

In the following example, we the list of endpoints from a specified container using C#.

```
/** This example makes use of the json-rpc-csharp project:
 *  https://github.com/adamashton/json-rpc-csharp
 */

String apiURL =
  "https://{domain}/api/v1.0/jsonrpc/";

// Make a request on the companies API.
Client rpcClient = new Client(apiURL + "network/computers");

String apiKey = "UjlMS+0m1l9IUZjpjWyJG8gbnv2Mta4T";
String userPassString = apiKey + ":";
String authorizationHeader = System.Convert.ToBase64String(
  System.Text.Encoding.UTF8.GetBytes(userPassString));

rpcClient.Headers.Add("Authorization",
  "Basic " + authorizationHeader);

JToken parameters = new JObject();
parameters["parentId"] = "55d43258b1a43ddf107baad4";
parameters["isManaged"] = True;
parameters["page"] = 1;
parameters["perPage"] = 2;

Request request = rpcClient.NewRequest(
  "getEndpointsList", parameters);

Response response = rpcClient.Rpc(request);
```

```
if (response.Result != null) {
    JToken result = response.Result;
    Console.WriteLine(response.ToString());
}
```

3.2. curl Example

In the following example, we get the list of containers for the mobile service in the Network API.

```
curl -i \
-H "Authorization: \
Basic VWpsTVMrMG0xbDlJVVpqcGpXeUpHOGdibnYyTXRhNFQ6" \
-H "Content-Type: application/json" \
-d '{"id": "123456789", "jsonrpc": "2.0",
"method": "getContainers", "params": []}' \
-X POST \
https://{domain}/api/v1.0/jsonrpc/network/mobile

HTTP/1.1 200 OK
Date: Wed, 10 Jan 2015 13:25:30 GMT
Content-Length: 103
Content-Type: application/json; charset=utf-8

{"id":"123456789","jsonrpc":"2.0","result":
  [{'id': '55d43258b1a43ddf107b23d8', 'name': 'Custom Groups'}]}
```

3.3. Python Example

Now, we will query the details of a company with Python.

```
import base64
import pyjsonrpc
import requests
import simplejson

# Generate Authorization header from API key
apiKey = "UjlMS+0m1l9IUZjpjWyJG8gbnv2Mta4T"
encodedUserPassSequence = base64.b64encode(apiKey + ":")
authorizationHeader = "Basic " + encodedUserPassSequence

json = pyjsonrpc.create_request_json("getPackagesList")
result = requests.post(
  "https://{domain}/api/v1.0/jsonrpc/packages",
  json,
  verify=False,
  headers = {
    "Content-Type": "application/json",
    "Authorization": authorizationHeader
  })

jsonResult = simplejson.loads(result.content)

print jsonResult

Output:

{'jsonrpc': '2.0',
 'id': '61f4dadc-bd10-448d-af35-16d45a188d9e',
 'result': {
 'items': [
 {'type': 3, 'id': '55d4325cb1a43ddf107b241b',
 'name': 'Default Security Server Package'},
 {'type': 4, 'id': '55d43e34b1a43db5187b23c6',
 'name': 'My package'}]
 , 'total': 2,
 'page': 1,
 'perPage': 30,
 'pagesCount': 0}
 }
```

3.4. Node.js example

In this example, we will make the exact previous call, only this time we will use Node.js

```
// Using the request module:
// npm install request
var request = require('request');

request({
  uri: "https://{domain}/ \
    api/v1.0/jsonrpc/packages",
  method: "POST",
  headers: {
    'Authorization':
      "Basic VWpsTVMrMG0xbDlJVVpqcGpXeUpHOGdibnYyTXRhNFQ6"
  },
  json: {
    "id": "123456789",
    "jsonrpc": "2.0",
    "method": "getPackagesList",
    "params": []
  }
}, function(response, body) {
  console.log(body);
});

// Output:

// {'jsonrpc': '2.0',
//   'id': '61f4dadc-bd10-448d-af35-16d45a188d9e',
//   'result': {
//   'items': [
//   {'type': 3, 'id': '55d4325cb1a43ddf107b241b',
//   'name': 'Default Security Server Package'},
//   {'type': 4, 'id': '55d43e34b1a43db5187b23c6',
//   'name': 'My package'}]
//   , 'total': 2,
//   'page': 1,
//   'perPage': 30,
//   'pagesCount': 0}
//   }
```