Bitdefender

GravityZone

API GUIDE

Bitdefender Control Center API Guide

Publication date 2021.07.06

Legal Notice

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.
6. **Quarantine**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/quarantine.
7. **General**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/general.
8. **Sandbox**, with the API URL:
   https://YOUR-HOSTNAME/api/v1.0/jsonrpc/sandbox.
9. **Sandbox Portal**, with the API URL:
   https://SANDBOX-IP/api/v1.

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

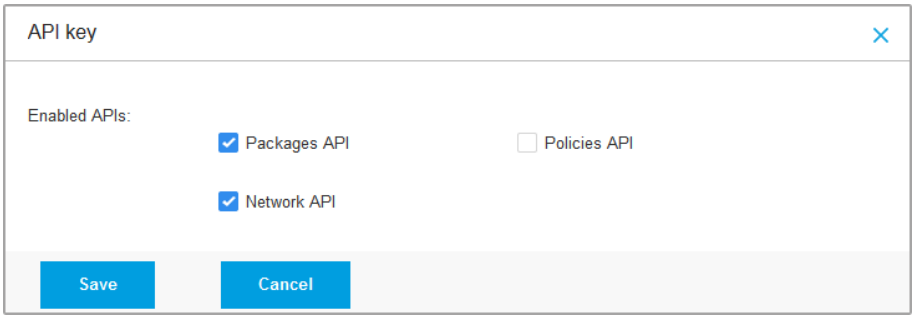
Note
Batch requests and notifications are not currently supported by Bitdefender Control Center.

1.3. API Keys

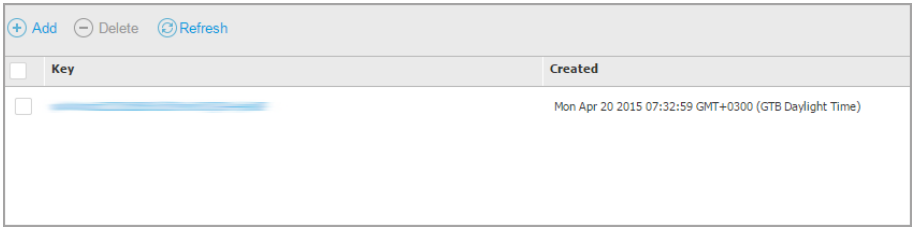
The API key is a unique key that is generated in **MyAccount** section of Bitdefender Control Center. Each API key allows the application to call methods exposed by one or several APIs. The allowed APIs are selected at the time the API key is generated.

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.



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



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 100 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	Yes	An object containing profile information: `fullName`, `timezone` and `language`.
`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`.
`targetIds`	Array	Yes	A list of IDs representing the targets to be managed by the user account.

Return value

This method returns a Boolean which is 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 `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

Parameter	Type	Optional	Description
			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 Notification Type and configurationSettings. .

Return value

This method returns a Boolean which is 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 no value is specified it will be set to its previous value or `False` if a aprevious value was not set.

Visibility option	Optional	Value
`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 syslog server, `False` otherwise. A syslog server must be configured in Control Center to receive this notification on the syslog server.

Visibility option	Optional	Value
		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 • `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.

Notification type	Available configurationSettings items with their type and possible values
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.
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.

Notification type	Available configurationSettings items with their type and possible values
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 • `reboot`, boolean, `True` to receive the notification when the Security Server needs a reboot, `False` otherwise

Notification type	Available configurationSettings items with their type and possible values
	• 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.
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 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.
38 - Blocked Devices	• deviceBlocked, array of integers between 0 and 21, representing the IDs of the device types: 1 - Bluetooth devices 2 - CD-ROM drives

Notification type	Available configurationSettings items with their type and possible values
	4 - Floppy disk drives 5 - IEEE 1284.4 6 - IEEE 1394 7 - Imaging devices 8 - Modems 10 - Tape drives 12 - Windows portable 13 - COM/LPT 14 - SCSI RAID 16 - Printers 18 - Wired network adapters 19 - Wireless network adapters 20 - Internal storage 21 - External storage

2.2. Network

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

- `getContainers` : returns the network containers.
- `getNetworkInventoryItems` : returns network inventory items.
- `createScanTask` : returns `true` if the task was successfully created.
- `createReconfigureClientTask` : creates a new Reconfigure Client task.
- `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.
- `setEndpointLabel` : sets a label to an endpoint.
- `createScanTaskByMac` : generates scan tasks for managed endpoints identified by MAC address.
- `assignPolicy` : this method is used to assign a policy template on the specified endpoints or containers.

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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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. getNetworkInventoryItems

This method returns network inventory items.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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 network items will be returned. If null, the items within the root custom group of the specified service are returned.

Parameter	Type	Optional	Description
`filters`	Object	Yes	The filters to be used when querying the endpoints list. For information regarding the available filters and how to use them, refer to "Available Filters" (p. 31).
`viewType`	Number	Yes	The ID of the view type. Each virtualization platform displays the inventory in specific views. 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	Number of items per page to be returned. The upper limit is 100 items per page. Default value: 30 items per page.

Available Filters

You can use the `filters` parameter to query the inventory items by certain properties. Filters are stuctured in sections and subsections, described hereinafter

The query result is a list of network items that match ALL sections AND subsections, AND ANY selected filter in a subsection.

These are the available filtering options:

Section	Subsection	Filtering Options
`type`		• `groups` - a Boolean to filter all custom groups of endpoints. Default value: `False`. This filter is available for `computers` service.

Section	Subsection	Filtering Options
		• `computers` - a Boolean to filter all computers. Default value: `False`. This filter is available for `computers` service. • `virtualMachines` - a Boolean to filter all virtual machines. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. • `clusters` - a Boolean to filter all Virtualization Clusters. Default value: `False`. This filter is available for `virtualmachines` service. • `hosts` - a Boolean to filter all Virtualization Hosts. Default value: `False`. This filter is available for `virtualmachines` service. • `dataCenters` - a Boolean to filter all Datacenters. Default value: `False`. This filter is available for `virtualmachines` service. • `vApps` - a Boolean to filter all vShield Apps. Default value: `False`. This filter is available for `virtualmachines` service. • `resourcePools` - a Boolean to filter all Resource Pools. Default value: `False`. This filter is available for `virtualmachines` service.

Section	Subsection	Filtering Options
		• `folders` - a Boolean to filter all Virtualization Folders. Default value: `False`. This filter is available for `virtualmachines` service. • `pools` - a Boolean to filter all Virtualization Pools. Default value: `False`. This filter is available for `virtualmachines` service.
`security`	`management`	• `managedWithBest` - a Boolean to filter all endpoints with the security agent installed on them. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. • `isContainerHost` - a Boolean to filter all endpoints with container host protection installed on them. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. • `managedExchangeServers` - a Boolean to filter all protected Exchange servers. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. This filter requires a valid license key that covers the Security for Exchange security service. • `managedRelays` - a Boolean to filter all endpoints with Relay role. Default value: `False`. This filter is available for `computers` and `virtualmachines` services.

Section	Subsection	Filtering Options
		• `securityServers` - a Boolean to filter all Security Servers. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. • `managedWithNsx` - a Boolean to filter protected endpoints in VMware NSX data centers. Default value: `False`. This filter is available for `virtualmachines` service. This filter requires a valid virtualization license key. • `managedWithVShield` - a Boolean to filter protected endpoints in VMware vShield environments. Default value: `False`. This filter is available for `virtualmachines` service. This filter requires a valid virtualization license key. • `managedWithHvi` - a Boolean to filter all endpoints managed by HVI. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. This filter requires a valid HVI license key.
`depth`		• `allItemsRecursively` - a Boolean to filter all endpoints recursively within the Network Inventory of a company. Default value: `False`.
`details`		• `ssid` - string, the SSID (Active Directory SID of the endpoint) used to filter the endpoints regardless of their protection status.

Section	Subsection	Filtering Options
		• macs - array, the list of MAC addresses used to filter the endpoints regardless of their protection status. • name - a String for filtering the items by name. Minimum required string length is three characters.

Important
Some filters require a specific license to be active, otherwise they are ignored, resulting in an inaccurate API response.
The field name works with partial matching.
The filter returns the items whose names are exact match or start with the specified value. To use the specified value as a suffix, use the asterisk symbol (*).
For example:
If name is computer, the API returns all items whose names start with computer.
If name is *puter, then the API returns a list of all items that contain puter in their names.

Return value

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

- page - the current page
- pagesCount - the total number of pages
- perPage - the total number of items returned per page
- total - the total number of items
- items - an array containing the list of items. Each entry in the list has the following fields:
  - id, the ID of the network item,
  - name, the name of the network item,
  - parentId, the ID of the parent container,
  - type, the type of network item: 4 - Group, 5 - Computer, 6 - Virtual Machine, 8 - Virtualization Host, 9 - vShield App, 10 - Virtualization Cluster, 11 - Virtualization Datacenter, 12 - Resource Pool, 13 - Virtualization Pool, 14 - Containers Group, 15 - Container Host Folder, 16 - Container

- `details`, more information about the item. This field is available for `5` - Computers, `6` - Virtual Machines and `16` - Containers. For information regarding the content of the details member please refer to "The details member" (p. 36). ,

The details member

Some network inventory items contain a `details` member. This member exposes more information regarding the item. The information depends on the item type.

Item type	Details
`5(computer)` and `6(virtual machine)`	• `label`, the label set to the endpoint • `fqdn`, the FQDN of the endpoint • `groupId`, the group ID of the endpoint • `isManaged`, boolean `True`, if this endpoint is managed • `machineType`, the type of the machine: (`1` - computer, `2` - virtual machine, `0` - Other) • `operatingSystemVersion`, the OS version of the endpoint • `ip`, the IP address of the endpoint • `macs`, the list of MAC addresses of the endpoint • `ssid`, the Active Directory SID of the endpoint • `managedWithBest`, boolean `True`, if BEST is installed on this endpoint • `isContainerHost`, boolean `True`, if this endpoint is a Container Host • `managedExchangeServer`, boolean `True`, if this endpoint is an Exchange Server • `managedRelay`, boolean `True`, if this endpoint has Relay role • `securityServer`, boolean `True`, if this endpoint is a Security Server • `managedWithNsx`, boolean `True`, if this is an endpoint from a VMware NSX data center • `managedWithVShield`, boolean `True`, if this is an endpoint from a VMware vShield environment

Item type	Details
	• managedWithHvi, boolean True, if this endpoint is managed by HVI

Example

Request :

```
{
     "params": {
         "parentId": "23b19c39b1a43d89367b32ce",
         "page": 2,
         "perPage": 5,
         "filters": {
             "type": {
                 "computers": true
             },
             "depth": {
                 "allItemsRecursively": true
             }
         }
     },
     "jsonrpc": "2.0",
     "method": "getNetworkInventoryItems",
     "id": "301f7b05-ec02-481b-9ed6-c07b97de2b7b"
}
```

Response :

```
{
    "id":"103d7b05-ec02-481b-9ed6-c07b97de2b7a",
    "jsonrpc":"2.0",
    "result": {
         page: 2,
         pagesCount: 11,
         perPage: 2,
         total: 22
         items[
             {
```

```
                    "id" : "21a295eeb1a43d8b497b23b7",
                    "name" : "Computer 21",
                    "type" : 2,
                    "parentId": "21a295eeb1a43d8b497b22b7",
                    "details" : {
                        "label" : "endpoint 1",
                        "fqdn": "endpoint1.local",
                        "groupId": "5a5f4d36b1a43d5f097b23bb",
                        "isManaged": true,
                        "machineType": 2,
                        "operatingSystemVersion": "Windows Server",
                        "ip": "60.40.10.220",
                        "macs": [
                            "324935237335"
                        ],
                        "ssid": "",
                    }
                },
                {
                    "id" : "21a295eeb1a43d8b497b24b7",
                    "name" : "Computer 22",
                    "type" : 2,
                    "parentId": "21a295eeb1a43d8b497b23b7",
                    "details" : {
                        "label" : "endpoint 2",
                        "fqdn": "endpoint2.local",
                        "groupId": "5a5f4d36b1a43d5f097b23bb",
                        "isManaged": true,
                        "machineType": 2,
                        "operatingSystemVersion": "Windows Server",
                        "ip": "60.40.10.220",
                        "macs": [
                            "324935237346"
                        ],
                        "ssid": "",
                    }
                }
            ]
        }
    }
```

2.2.3. 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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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; `4` - custom scan
`name`	String	Yes	The name of the task. If the parameter is not passed, the name will be automatically generated.
`customScanSettings`	Array	No	Object containing information such as scan depth and scan path(s). This object should be set only when `type` parameter has the value `4` - Custom scan. When set for other types, the values will be ignored. Parameter

Parameter	Type	Optional	Description
			$customScanSettings must contain the following properties: int $scanDepth The scan profile. Available options: 1 - aggressive; 2 - normal; 3 - permissivearray $scanPath The list of target paths to be scanned

Return value

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

Example

Request :

```
{
      "params": {
          "targetIds": ["559bd17ab1a43d241b7b23c6",
                        "559bd17ab1a43d241b7b23c7"],
          "type": 4,
          "name": "my scan",
          "customScanSettings": {
              "scanDepth": 1,
              "scanPath": [
                  "LocalDrives"
              ]
          }
      },
      "jsonrpc": "2.0",
      "method": "createScanTask",
      "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

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

2.2.4. createReconfigureClientTask

This method creates a new Reconfigure Client task. With this task you can choose which modules to install on target agents.

Warning
The `networkMonitor` module is deprecated. It is recommended to use `networkAttackDefense` instead.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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	The endpoint or container IDs, for which you want to reconfigure the agents.
`scheduler`	Object	Yes	The task scheduler settings. The object contains the following fields: • `type`, an Integer with one of the following values: – 1 for immediate run (default)

Parameter	Type	Optional	Description
			– 2 for scheduled If `type` is 1, you can omit the other fields. • `recurrence`, an Integer with one of the following values: – 1 for hourly. This value requires `everyHour` to be set. – 2 for daily. This value requires `startTime` to be set. – 3 for weekly. This value requires both `everyHour` and `startTime` to be set. • `everyHour`, an Integer between 1 and 23, representing the interval in hours between two task runs. • `startTime`, a string with the following format: `HH:mm`, representing the hour of the first task run. • `onWeekDay`, an Integer between 1 and 7, where 1 is Monday and 7 is Sunday. If this parameter is omitted, the task runs immediately.
`modules`	Object	Yes	The modules to be enabled or disabled. The object contains the following fields: • `antimalware` • `advancedThreatControl` • `firewall` • `contentControl` • `deviceControl` • `powerUser` • `applicationControl` • `encryption` • `advancedAntiExploit` • `containerProtection`

Parameter	Type	Optional	Description
			• `edrSensor` • `patchManagement` • `networkAttackDefense` Each field may have the value 1 for enabled, or 0 for disabled. If the module is omitted, it is considered disabled.
`scanMode`	Object	Yes	The settings for the scanning engines. The object contains the following fields: • `type`, an Integer with one of the following values: – 1 for automatic configuration (default) – 2 for custom settings. This value requires the `computers` and `vms` fields If omitted, the default values will be used. • `vms`, an Object described below. • `computers`, an Object described below. The objects `computers` and `vms` have the following fields: • `main`, an Integer with one of the following values: – 1 for Central Scanning (with Security Server) – 2 for Hybrid Scanning (light engines) – 3 for Local Scanning (full engines) • `fallback`, an Integer with one of the following values: – 2 for Hybrid Scanning (light engines) – 3 for Local Scanning (full engines) If `main` has the value 2 or 3, then `fallback` is not considered.

Parameter	Type	Optional	Description
`roles`	Object	Yes	The roles to be enabled or disabled on the agent: • `relay` with the following possible values: – 1 for enabled – 0 for disabled (default) • `exchange` with the following possible values: – 1 for enabled – 0 for disabled (default) This role is available only with a valid Security for Exchange license.
`productType`	Number	Yes	This parameter determines the operation mode of the security agent. Possible values: • `0` - for Detection and prevention mode, default for full endpoint security agents. • `3` - for EDR (Report only) mode, default for Bitdefender EDR agents. For additional information, refer to "Parameter Info" (p. 44).

Parameter Info

- Bitdefender EDR users can only run tasks that reconfigure target security agents to operate in EDR (Report only) mode; specifying `productType` is optional.
- GravityZone BS / ABS / Elite and Enterprise users can only run tasks that reconfigure target security agents to operate in Detection and prevention mode; specifying `productType` is optional.
- GravityZone Ultra users can reconfigure target security agents to operate in both operation modes.
  - `productType` must be specified for EDR (Report only) mode reconfiguration.
  - In case of selecting endpoints running different operation modes, if `productType` is not specified, the EDR (Report only) endpoints will be reconfigured to run in Detection and prevention mode.

- The EDR (Report only) mode includes by default a set of predefined parameters that will overwrite user-specified options. Predefined parameters:
  - modules
    - edrSensor - true
    - contentControl - true
    - networkAttackDefense - true
    - advancedThreatControl - true
    - other modules - false
  - scanMode - n/a
  - roles.exchange - false

Return value

This method returns a Boolean which is True if the reconfigure task was created successfully for at least one target ID.

Example

Request :

```
{
   "params": {
       "targetIds": [
           "5d7244b10ea1de153817c072"
       ],
       "scheduler": {
           "type": 1
       },
       "modules": {
         "advancedThreatControl": 1,
         "firewall": 1,
         "contentControl": 1,
         "deviceControl": 1,
         "powerUser": 1,
         "encryption": 1,
         "advancedAntiExploit": 1,
         "containerProtection": 1,
         "edrSensor": 1,
         "patchManagement": 1,
         "applicationControl": 1,
```

```
            "networkAttackDefense":1
        },
            "scanMode": {
            "type": 1
        },
        "roles": {
            "relay": 0,
            "exchange": 0
        },
        "productType": 0
    },
    "jsonrpc": "2.0",
    "method": "createReconfigureClientTask",
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

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

2.2.5. getScanTasksList

This method returns the list of scan tasks.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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.
`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 100 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'
             },
         ]
    }
}
```

2.2.6. 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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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	Yes	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.

Parameter	Type	Optional	Description
			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 100 items per page. Default value: 30 items per page.
`filters`	Object	Yes	The filters to be used when querying the endpoints list. For information regarding the available filters and how to use them, refer to "Available Filters" (p. 50).

Available Filters

You can use the `filters` parameter to query the endpoints by certain properties. Filters are stuctured in sections and subsections, described hereinafter

The query result is a list of endpoints that match ANY selected filter in ALL sections AND subsections.

These are the available filtering options:

Section	Subsection	Filtering Options
`security`	`management`	• `managedWithBest` - a Boolean to filter all endpoints with the security agent installed on them. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. • `managedExchangeServers` - a Boolean to filter all protected Exchange servers. Default value: `False`.

Section	Subsection	Filtering Options
		This filter is available for `computers` and `virtualmachines` services. This filter requires a valid license key that covers the Security for Exchange security service. • `managedRelays` - a Boolean to filter all endpoints with Relay role. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. • `securityServers` - a Boolean to filter all Security Servers. Default value: `False`. This filter is available for `computers` and `virtualmachines` services. • `managedWithNsx` - a Boolean to filter protected endpoints in VMware NSX data centers. Default value: `False`. This filter is available for `virtualmachines` service. This filter requires a valid virtualization license key. • `managedWithVShield` - a Boolean to filter protected endpoints in VMware vShield environments. Default value: `False`. This filter is available for `virtualmachines` service. This filter requires a valid virtualization license key. • `managedWithHvi` - a Boolean to filter all endpoints managed by HVI. Default value: `False`. This filter is available for `computers` and `virtualmachines` services.

Section	Subsection	Filtering Options
		This filter requires a valid HVI license key.
depth		• allItemsRecursively - a Boolean to filter all endpoints recursively within the Network Inventory of a company. Default value: False.
details		• ssid - string, the SSID (Active Directory SID of the endpoint) used to filter the endpoints regardless of their protection status. • macs - array, the list of MAC addresses used to filter the endpoints regardless of their protection status. • name - string, used for filtering the endpoints by name regardless of their protection status. Minimum required string length is three characters.

Important
Some filters require a specific license to be active, otherwise they are ignored, resulting in an inaccurate API response.
The field name works with partial matching.
The filter returns the endpoints whose names are exact match or start with the specified value. To use the specified value as a suffix, use the asterisk symbol (*).
For example:
If name is computer, the API returns all endpoints whose names start with computer.
If name is *puter, then the API returns a list of all endpoints that contain puter in their names.

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`, the ID of managed endpoint,
  - `name`, the name of the endpoint,
  - `label`, the label set to this endpoint,
  - `fqdn`, the FQDN of the endpoint,
  - `groupId`, the group ID of the endpoint,
  - `isManaged`, boolean `True`, if this endpoint is managed,
  - `machineType`, the type of the machine: (1 - computer, 2 - virtual machine, 3 - EC2 Instance, 5 - container host, 0 - Other),
  - `operatingSystemVersion`, the operating system version of the endpoint,
  - `ip`, the IP address of the endpoint,
  - `macs`, the MAC addresses of the endpoint,
  - `ssid`, the SSID (Active Directory SID) of the endpoint,
  - `managedWithBest`, boolean `True`, if BEST is installed on this endpoint,
  - `isContainerHost`, boolean `True`, if this endpoint is a Container Host,
  - `managedExchangeServer`, boolean `True`, if this endpoint is an Exchange Server,
  - `managedRelay`, boolean `True`, if this endpoint has Relay role,
  - `securityServer`, boolean `True`, if this endpoint is a Security Server,
  - `managedWithNsx`, boolean `True`, if this is an endpoint from a VMware NSX data center,
  - `managedWithVShield`, boolean `True`, if this is an endpoint from a VMware vShield environment,
  - `managedWithHvi`, boolean `True`, if this endpoint is managed by HVI,
  - `hviProtectionType`, the type of the HVI protection (1 - Security Server Multi-platform, 2 - BEST)

Example

Request :

```
{
      "params": {
          "parentId": "23b19c39b1a43d89367b32ce",
```

```
            "page": 2,
            "perPage": 5,
            "filters": {
                "security": {
                    "management": {
                        "managedWithBest": true,
                        "managedRelays": true
                    }
                }
            }
        },
        "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",
                 "label" : "endpoint 1",
                 "fqdn": "endpoint1.local",
                 "groupId": "5a5f4d36b1a43d5f097b23bb",
                 "isManaged": true,
                 "machineType": 2,
                 "operatingSystemVersion": "Windows Server 2016",
                 "ip": "60.40.10.220",
                 "macs": [
                          "324935237335"
                 ],
```

```
                    "ssid": "",
                    "managedWithHvi": true,
                    "hviProtectionType": 1,
                    "managedWithBest": true

                },
                {
                    "id" : "23a295d8b1a43d7c4a7b23c9",
                    "name" : "Endpoint 2",
                    "machineType": 2,
                    "label" : "endpoint 2",
                    "fqdn": "endpoint2.local",
                    "groupId": "5a4f4d46b1a53d5f197b23aa",
                    "isManaged": true,
                    "machineType": 2,
                    "operatingSystemVersion": "Windows 7",
                    "ip": "60.40.10.221",
                    "macs": [
                            "325935237445"
                    ],
                    "ssid": "",
                    "managedWithHvi": true,
                    "hviProtectionType": 1

                }
            ]
        }
    }
```


2.2.7. getManagedEndpointDetails

This method returns detailed information, such as: details to identify the endpoint and the security agent, the status of installed protection modules.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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
- `companyId` - the ID of the company to which the endpoint belongs
- `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 address 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 containing the following information about the agent installed on the endpoint:
  - `engineVersion`, string, the version of the scanning engine
  - `primaryEngine`, the first engine to be used when scanning for malware. It can have one of the following values:
    - `1` - for Central Scanning (Security Server)
    - `2` - for Hybrid Scanning (Light Engines)
    - `3` - for Local Scanning (Full Engines)

- 0 - Unknown

- fallbackEngine, the engine to be used if the primary engine is unavailable when the task is sent. It can have one of the following values:
  - 2 - for Hybrid Scanning (Light Engines)
  - 3 - for Local Scanning (Full Engines)
  - 0 - Unknown
- lastUpdate, the time and date of the last signatures update
- licensed, integer, the license status: 0 - pending authentication, 1 - active license, 2 - expired license, 6 - there is no license or not applicable
- productOutdated, a Boolean specifying whether the agent's version is the latest available or not
- productUpdateDisabled, a Boolean specifying if product updates are disabled
- productVersion, string, the version of the agent
- signatureOutdated, a Boolean specifying if the antimalware signatures of the endpoint are outdated
- signatureUpdateDisabled, a Boolean specifying if the antimalware signature updates are disabled
- type, identifies which type of agent is installed on the endpoint:
  - 1 - Endpoint Security
  - 2 - Bitdefender Tools
  - 3 - BEST

- group - an Object pointing to the group to which the endpoint belongs. The object contains the following fields:
  - id, the ID of the group
  - name, the name of the group
- malwareStatus - an Object informing of the status of the endpoint related to malware. The object has the following fields:

  - `detection`, a Boolean indicating if malware was detected on the endpoint in the last 24 hours,
  - `infected`, a Boolean informing if the antimalware was able to remove the infection or the endpoint is still infected
- `policy` - an Object informing about the active policy on the endpoint. The object contains:
  - `id`, the ID of the active policy,
  - `name`, the name of the policy,
  - `applied`, a Boolean set to True if the policy is currently applied on the endpoint
- `hypervisorMemoryIntrospection` - an Object providing the status and configuration of Bitdefender HVI. This object appears only if the endpoint is protected by HVI.

  Object description:
  - `status`, a Boolean set to True if HVI is enabled
  - `activeModules`, an Object containing two Boolean fields that show the status of the HVI modules: `userMode` and `kernelMode`. If True, then the module is active.
  - `securityServer`, an Object that contains the details 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` - an Object informing of the installed modules and their statuses. The fields have Boolean values, `True` - if the module is enabled, or `False` - if the module is disabled.

  The available fields are:
  - `advancedThreatControl`
  - `antimalware`

  - `contentControl`
  - `deviceControl`
  - `firewall`
  - `powerUser`
  - `encryption`
  - `hyperDetect`
  - `patchManagement`
  - `relay`
  - `exchange`
  - `sandboxAnalyzer`
  - `advancedAntiExploit`.
  - `containerProtection`.
  - `edrSensor`.
  - `networkAttackDefense`.
- `label` - string, the label set to this endpoint
- `managedWithBest` - a Boolean set to True if the agent (BEST) is installed on the endpoint.
- `isContainerHost` - a Boolean set to True if the endpoint is a Container Host.
- `managedExchangeServer` - a Boolean set to True if the endpoint is an Exchange Server
- `managedRelay` - a Boolean set to True if the endpoint has Relay role
- `securityServer` - a Boolean set to True if the endpoint is a Security Server
- `managedWithNsx` - a Boolean set to True if the endpoint is in a protected VMware NSX data center
- `managedWithVShield` - a Boolean set to True if the endpoint is in a protected VMware vShield environment

- `managedWithHvi` - a Boolean set to True if the endpoint si protected by Bitdefender HVI
- `hviProtectionType` - informs how HVI protection is delivered: 1 - via Security Server, 2 - via agent (BEST)

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',
        'companyId': '5575a235d2172c65038b454e',
        '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,
            'networkAttackDefense': 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'
        },
        "label" : "endpoint label",
```

```
            "managedWithHvi": true,
            "hviProtectionType": 1,
            "managedWithBest": true
        }
    }
```

2.2.8. createCustomGroup

This method creates a new custom group.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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"
}
```

2.2.9. deleteCustomGroup

This method deletes a custom group.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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
}
```

2.2.10. moveCustomGroup

This method moves a custom group to another custom group.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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
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
}
```

2.2.11. moveEndpoints

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

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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
}
```

2.2.12. 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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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
}
```

2.2.13. setEndpointLabel

This method sets a new label to an endpoint.

Parameters

Parameter	Type	Optional	Description
`endpointId`	String	No	The endpoint ID.
`label`	String	No	A string representing the label. The maximum allowed length is 64 characters. Enter an empty string to reset a previously set label.

Return value

This method returns a Boolean which is True, when the label was successfully set.

Example

Request :

```
{
      "params": {
          "endpointId": "5a30e7730041d70cc09f244b",
          "label": "label with url http://test.com?a=12&b=wow"
      },
      "jsonrpc": "2.0",
      "method": "setEndpointLabel",
      "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

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

2.2.14. createScanTaskByMac

Use this method to generate a scan task for managed endpoints identified by their MAC address.

Parameters

Parameter	Type	Optional	Description
`macAddresses`	Array	No	The list of mac addresses of the endpoints to be scanned. You can specify at most 100 MAC addresses at once
`type`	Number	No	The type of scan. Available options: `1` - quick scan; `2` - full scan; `3` - memory scan; `4` - custom scan
`name`	String	Yes	The name of the task. If the parameter is not passed, the name will be generated automatically.

Parameter	Type	Optional	Description
customScanSettings	Array	No	Object containing information such as scan depth and scan path(s). This object should be set only when type parameter has the value 4 - Custom scan. When set for other types, the values will be ignored. Parameter $customScanSettings must contain the following properties: int $scanDepth The scan profile. Available options: 1 - aggressive; 2 - normal; 3 - permissivearray $scanPath The list of target paths to be scanned

Return value

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

Example

Request :

```
{
      "params": {
          "macAddresses": [
              "1c67da49e1a1",
              "8c67f849e1a8"
          ],
          "type": 4,
          "name": "my scan",
          "customScanSettings": {
              "scanDepth": 1,
              "scanPath": [
                  "LocalDrives"
              ]
          }
      },
```

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

Response :

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

2.2.15. assignPolicy

This method assigns a policy to a list of endpoints or containers.

Note
The method uses the default view type. For VMWare integrations it is **Hosts and Clusters** view. For Citrix XenServer integrations it is **Server** view. If you are using other views, you must include in `targetIds` the IDs of the target endpoints and containers.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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 with the IDs of the policy targets. The target ID can designate an endpoint or a container.
`inheritFromAbove`	Boolean	Yes	A boolean specifying whether the given targets shoud inherit the policy of the parent container. Targets without a parent container receive the default policy. Use this parameter only in conjuction with the `targetIds` parameter. By default, the parameter is set to `True`.
`policyId`	String	Yes	A string specifying the ID of the policy to be assigned. When this parameter is missing the `inheritFromAbove` parameter must be set to `True`.
`forcePolicyInheritance`	Boolean	Yes	A boolean specifying whether the policy should be assigned to child entities of the given targets. By default, the parameter is set to `False`.

Return value

This method returns a Boolean which is True, when the policy was successfully assigned to one or more targets. The policy is not assigned to targets that have enforced policy.

Example

Request :

```
{
      "params": {
          "targetIds": [
              "56728d66b1a43de92c712346",
              "69738d66b1a43de92c712346"
          ],
          "inheritFromAbove": false,
          "policyId": "55828d66b1a43de92c712345",
          "forcePolicyInheritance": true
      },
      "jsonrpc": "2.0",
      "method": "assignPolicy",
      "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f"
}
```

Response :

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

2.3. Packages

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

- `getInstallationLinks` : returns the installation links and full kits for a package.
- `getPackagesList` : returns the list of available packages.
- `createPackage` : creates a new package and returns its ID.
- `deletePackage` : deletes a package.
- `getPackageDetails` : retrieves information about a package.

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

2.3.1. getInstallationLinks

This method returns the installation links and full kits for a package and their availability status.

Warning
Make sure you have the right access permissions and the installation package is published in the `Configuration > Update > Components` page of Control Center. Otherwise, you may receive error `400 Bad request` or `404 Not found`.

Parameters

Parameter	Type	Optional	Description
`packageName`	String	Yes	The name of the package. If no value is passed, all packages will be returned.

Return value

This method returns an Array containing the list of installation links for the requested package, or for all available packages if none specified explicitly. Each entry in the list has the following fields:

- `packageName` - the name of the package for which you need the installation links and kits
- `installLinkWindows` - the installation link for Windows operating systems
- `installLinkMac` - the installation link for macOS operating systems
- `installLinkLinux` - the installation link for Linux operating systems
- `fullKitWindowsX32` - the full kit for Windows x32 operating systems
- `fullKitWindowsX64` - the full kit for Windows x64 operating systems
- `fullKitLinuxX32` - the full kit for Linux x32 operating systems
- `fullKitLinuxX64` - the full kit for Linux x64 operating systems
- `status` - an Object containing the supported operating systems and showing kits availability within your GravityZone environment. Possible status values: 0 - not downloaded, 1 - downloading, 2 - ready
  - `windows`

- linux
- mac

Example

Request :

```
{
   "params": {
     "packageName": "my package"
 },
   "jsonrpc": "2.0",
   "method": "getInstallationLinks",
   "id": "426db9bb-e92a-4824-a21b-bba6b62d0a18"
}
```

Response :

```
{
   "id": "426db9bb-e92a-4824-a21b-bba6b62d0a18",
   "jsonrpc": "2.0",
   "result": [{
       "packageName": "Pack1",
       "installLinkWindows":
 "https://CONTROL_CENTER_APIs_ACCESS_URL/Packages/BSTWIN/0/ \
  setupdownloader_[qwer=].exe",
       "installLinkMac":
 "https://CONTROL_CENTER_APIs_ACCESS_URL/Packages/MAC/0/ \
  antivirus_for_mac_[qwer].pkg",
       "installLinkLinux":
 "https://CONTROL_CENTER_APIs_ACCESS_URL/Packages/BSTNIX/0/ \
  0E_rWP/installer",
       "fullKitWindowsX32":
 "https://YOUR-HOSTNAME/api/v1.0/http/downloadPackageFullKit? \
  packageId=5f1ecde1be4be6142c3e9b32&downloadType=19",
       "fullKitWindowsX64":
 "https://YOUR-HOSTNAME/api/v1.0/http/downloadPackageFullKit? \
  packageId=5f1ecde1be4be6142c3e9b32&downloadType=20",
       "fullKitLinuxX32":
 "https://YOUR-HOSTNAME/api/v1.0/http/downloadPackageFullKit? \
```

```
  packageId=5f1ecde1be4be6142c3e9b32&downloadType=21",
       "fullKitLinuxX64":
 "https://YOUR-HOSTNAME/api/v1.0/http/downloadPackageFullKit? \
  packageId=5f1ecde1be4be6142c3e9b32&downloadType=22",
       "status": {
         "windows": 0,
         "linux": 1,
         "mac": 2
    }]
}
```

Request :

```
Download the full kit package using curl:

curl -fOJ -H "YOUR_API_KEY:" \
https://YOUR-HOSTNAME/api/v1.0/http/\
downloadPackageFullKit?packageId=5645cba6f12a9a8c5e8b4748&\
downloadType=20

Equivalent with:

curl -fOJ -H "Authorization: Basic API_KEY_ENCODED_BASE64" \
https://YOUR-HOSTNAME/api/v1.0/http/\
downloadPackageFullKit?packageId=5f1ecde1be4be6142c3e9b32&\
downloadType=20

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

2.3.2. 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 100 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. The type can have the following values:
  - 3 Security Server
  - 4 Bitdefender Endpoint Security Tools
  - 5 Sandbox Analyzer
  - 6 Network Sensor appliance for Sandbox Analyzer

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
             }
         ]
    }
}
```

2.3.3. createPackage

This method creates an installation package.

Warning
The `atc` module is deprecated. It is recommended to use `advancedThreatControl` instead.

Parameters

Parameter	Type	Optional	Description
`packageName`	String	No	The name of the package.

Parameter	Type	Optional	Description
`description`	String	Yes	The description of the package. If no value is passed, the description will be an empty string.
`language`	String	Yes	The language of the package in the LL_CC format, where LL is the language and CC is the country. The supported languages are: en_US, es_ES, de_DE, fr_FR, ro_RO, pl_PL, pt_BR, it_IT, ru_RU. If not specified, the default value is `en_US`.
`modules`	Object	Yes	An object with the modules to be enabled/disabled. The keys can be: • `advancedThreatControl`, • `firewall`, • `contentControl`, • `deviceControl`, • `powerUser`, • `containerProtection`, • `applicationControl`, • `advancedAntiExploit`, • `encryption`, • `patchManagement`, • `edrSensor`, • `networkAttackDefense`. The values can be 1 (enabled) or 0 (disabled). If the module is not sent, it will be considered disabled.
`scanMode`	Object	Yes	An object with the scan mode settings. Object description: • The accepted keys are: `type`, `vms` and `computers`. The `type` value

Parameter	Type	Optional	Description
			can be 1 (automatic) or 2 (for custom mode). • If `type` value is 2, then the `computers` respectively `vms` keys and values need to be sent, otherwise the default values will be filled by the system. The value for `computers` or `vms` is an object with the possible keys: `main` and `fallback`. • The values for `main` can be 1 (for Central Scanning (Security Server)), 2 (for Hybrid Scanning (Light Engines)) or 3 (for Local Scanning (Full Engines)). • The values for `fallback` can be 2 (for Hybrid Scanning (Light Engines)) or 3 (for Local Scanning (Full Engines)). If the value for `main` option is 2 or 3, the value of `fallback` will not be considered. • If this parameter is not sent, the values for automatic mode are saved.
`settings`	Object	Yes	An object with other settings of the package. The values can be: • `scanBeforeInstall`, • `removeCompetitors`, • `uninstallPassword`, • `customInstallationPath`, • `customGroupId`, • `vmsCustomGroupId`.

Parameter	Type	Optional	Description
			The value for `scanBeforeInstall` can be 1 (enabled) or 0 (disabled). The value for `removeCompetitors` can be 1 (enabled) or 0 (disabled). `uninstallPassword` should be a string and it should meet the complexity requirements: The password must be at least 6 characters in length and it must contain at least one digit, one upper case, one lower case and one special character; and `customInstallationPath` should be a valid Windows path where the package will be installed (this will work only for Windows operating systems). `customGroupId` should be a string representing the ID of the custom group entity where the new endpoint should be deployed for the Computers and Virtual Machines View. `vmsCustomGroupId` should be a string representing the ID of the custom group entity where the new endpoint should be deployed for the Virtual Machines View. All values are optional.
`roles`	Object	Yes	An object containing the roles to be enabled or disabled: • `relay` with the following possible values: `1` for enabling the Relay role, and `0` to disable it. By default, the Relay role is disabled. • `exchange` with the following possible values: `1` for enabling the Exchange role, and `0` to disable it.

Parameter	Type	Optional	Description
			By default, the Exchange role is disabled. This role is available only with a valid Security for Exchange license.
`deploymentOptions`	Object	Yes	An object containing installation options: • `type`, an integer indicating the entity to which the endpoint will connect to. This entity will deliver the installation kit and updates. Possible values are: `1` for regular deploy from the Bitdefender Update Server; `2` for deployments through a Relay. • `relayId`, a string representing the ID of an endpoint with the Relay role enabled. This field must be set when the `type` option is set to `2`, meaning deploying using a Relay. • `useCustomCommunicationServer`, a boolean allowing you to choose if the endpoint will communicate with a specific Communication Server. Possible values are: `True` to specify a specific Communication Server, `False` to use the default Communication Server. This option may be set when the deploy option is `1`, meaning regular deploy. • `communicationServer`, a string containing the IP or hostname of the custom Communication Server. This option must be set only when

Parameter	Type	Optional	Description
			`useCustomCommunicationServer` is set to `True`. • `useCommunicationProxy`, a boolean allowing you to specify if the endpoint will use a proxy to communicate over the Internet. Possible values are: `True` to use a communication proxy, `False` otherwise. • `proxyServer`, a string representing the IP or domain name of the proxy server. Valid values are IP addresses in IPV4 or IPV6 format and domain names as defined under RFC 1035. This option is required when `useCommunicationProxy` is set to `True`. • `proxyPort`, an integer representing the port which allows access to the proxy server. Valid values are between `1` and `65535`. This option is required when `useCommunicationProxy` is set to `True`. • `proxyUsername`, a string representing the username required for authentication with the proxy server. This option may be omitted if the proxy server does not require authentication. • `proxyPassword`, a string representing the password required for authentication on the proxy server. This option may be omitted

Parameter	Type	Optional	Description
			if the proxy server does not require authentication.
`productType`	Number	Yes	This parameter determines the operation mode of the security agent. Possible values: • `0` - for Detection and prevention mode, default for full endpoint security agents. • `3` - for EDR (Report only) mode, default for Bitdefender EDR agents. For additional information, refer to "Parameter Info" (p. 84).

Parameter Info

- Bitdefender EDR users can only create EDR (Report only) packages; specifying `productType` is optional.
- GravityZone BS / ABS / Elite and Enterprise users can only create Detection and prevention packages; specifying `productType` is optional.
- GravityZone Ultra users can create both EDR (Report only) and Detection and prevention packages; `productType` must be specified to create an EDR (Report only) package.
- The EDR (Report only) package includes by default a set of predefined parameters that will overwrite user-specified options. Predefined parameters:
  - `modules`
    - `edrSensor - true`
    - `contentControl - true`
    - `networkAttackDefense - true`
    - `advancedThreatControl - true`
    - `other modules - false`
  - `scanMode - n/a`
  - `settings.removeCompetitors - false`

- settings.scanBeforeInstall - false
- roles.exchange - false

Return value

This method returns an Array containing an object with the ID of the created package and the status of the call, if successful.

Example

Request :

```
{
   "params": {
     "packageName": "a unique name",
     "companyId": "54a1a1d3b1a43d2b347b23c1",
     "description": "package description",
     "language": "en_EN",
     "modules": {
         "advancedThreatControl": 1,
         "firewall": 0,
         "contentControl": 1,
         "deviceControl": 0,
         "powerUser": 0,
         "containerProtection": 0,
         "applicationControl": 0,
         "advancedAntiExploit": 0,
         "encryption": 0,
         "patchManagement": 0,
         "edrSensor": 0,
         "networkAttackDefense": 0
     },
     "scanMode": {
         "type": 2,
         "computers": {
             "main": 1,
             "fallback": 2
         },
         "vms": {
             "main": 2
         }
     },
```

```
        "settings": {
            "uninstallPassword": "mys3cre3tP@assword",
            "scanBeforeInstall": 0,
            "removeCompetitors": 1,
            "customInstallationPath": "c:\\mypath\\bitdefender",
            "customGroupId": "5a4dff50b1a43ded0a7b23c8",
            "vmsCustomGroupId": "5a4dff50b1a43ded0a7b23c7"
        },
        "roles": {
            "relay": 0,
            "exchange": 1
        },
        "deploymentOptions": {
            "type": 2,
            "relayId": "54a1a1s3b1a43e2b347s23c1",
            "useCommunicationProxy": true,
            "proxyServer": "10.12.13.14",
            "proxyPort": 123
        },
        "productType": 0
    },
    "jsonrpc": "2.0",
    "method": "createPackage",
    "id": "426db9bb-e92a-4824-a21b-bba6b62d0a18"
}
```

Response :

```
{
    "id": "426db9bb-e92a-4824-a21b-bba6b62d0a18",
    "jsonrpc": "2.0",
    "result": [
        {
          "records": ["551bb0aed5172cac5c8b4568"],
          "success": true
        }
    ]
}
```

2.3.4. deletePackage

This method deletes a package identified through the provided package ID.

Parameters

Parameter	Type	Optional	Description
packageId	String	No	The ID of the package to be deleted.

Return value

This method does not return any value.

Example

Request :

```
{
  "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
  "jsonrpc": "2.0",
  "method": "deletePackage",
  "params": {
       "packageId": "5a37b660b1a43d99117b23c6"
      }
}
```

Response :

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

2.3.5. getPackageDetails

This method retrieves information about the configuration of a specific package identified through the provided ID.

Warning
The `atc` module is deprecated and it will be removed from API in the near future.

Parameters

Parameter	Type	Optional	Description
`packageId`	String	No	The ID of the package for which details should be retrieved.

Return value

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

- `packageName` - the name of the package.
- `description` - the description of the package.
- `language` - the language of the package in the LL_CC format, where LL and CC are landugage and country international codes.
- `modules` - indicating the status of the modules present in the package. The object may contain the following members: `antimalware`, `advancedThreatControl` and `atc`, `firewall`, `contentControl`, `deviceControl`, `powerUser`, `containerProtection`, `applicationControl`, `advancedAntiExploit`, `encryption`, `patchManagement`, `edrSensor`, `networkAttackDefense`. The value for each module is either `1` (enabled) or `0` (disabled).
- `scanMode` - an object describing the scan mode settings and containing the following fields:
  - `type`, with the following values `1` (automatic) or `2` (for custom mode)
  - `computers`, an object with the possible fields: `main` for the main scanning engine and `fallback` for the fallback scanning engine. The values of these fields can be `1` - Central Scanning with Security Server, `2` - Hybrid Scanning (Light Engines) or `3` - Local Scanning (Full Engines)
  - `vms`, an object with the possible fields: `main` for the main scanning engine and `fallback` for the fallback scanning engine. The values of these fields

can be 1 - Central Scanning with Security Server, 2 - Hybrid Scanning (Light Engines) or 3 - Local Scanning (Full Engines)

- settings - an object with other settings of the package containing the following fields:
  - scanBeforeInstall,
  - removeCompetitors,
  - customInstallationPath,
  - customGroupId,
  - vmsCustomGroupId.
- roles - an object containing the enabled/disabled roles:
  - relay with the following possible values: 1 if enabled and 0 if disabled.
  - exchange with the following possible values: 1 if enabled, and 0 if disabled.
- deploymentOptions - an object containing installation options:
  - type, an integer indicating the entity to which the endpoint will connect to. This entity will deliver the installation kit and updates. Possible values are: 1 for regular deploy from the Bitdefender Update Server; 2 for deployments through a Relay.
  - relayId, a string representing the ID of an endpoint with the Relay role enabled. This field is returned if type option is set to 2, meaning deploying using a Relay.
  - useCustomCommunicationServer, a boolean specifying whether the endpoint communicates with a specific Communication Server.
  - communicationServer, a string containing the IP or hostname of the custom Communication Server. This option is returned only when useCustomCommunicationServer is set to True.
  - useCommunicationProxy, a boolean specifying whether the endpoint uses a proxy to communicate over the Internet. Possible values are: True to use a communication proxy, False otherwise.
  - proxyServer, a string representing the IP or domain name of the proxy server. Valid values are IP addresses in IPV4 or IPV6 format and domain names as defined under RFC 1035. This option is present when useCommunicationProxy is set to True.

  - `proxyPort`, an integer representing the port which allows access to the proxy server. Valid values are between `1` and `65535`. This option is present when `useCommunicationProxy` is set to `True`.
  - `proxyUsername`, a string representing the username required for authentication with the proxy server. This option may be omitted if the proxy server does not require authentication.
- `productType` - the assigned product type. This field determines the operation mode of the security agent. Possible values:
  - `0`, for Detection and prevention
  - `3`, for EDR (Report only)

Example

Request :

```
{
  "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
  "jsonrpc": "2.0",
  "method": "getPackageDetails",
  "params": {
      "packageId": "5a37b660b1a43d99117b23c6"
  }
}
```

Response :

```
{
  "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
  "jsonrpc": "2.0",
  "result": {
     "packageName": "Package",
     "description": "package description",
     "language": "en_US",
     "modules": {
         "antimalware": 1,
         "advancedThreatControl": 1,
         "atc": 1,
         "firewall": 0,
```

```
            "contentControl": 1,
            "deviceControl": 0,
            "powerUser": 0,
            "containerProtection": 0,
            "applicationControl": 0,
            "advancedAntiExploit": 0,
            "encryption": 0,
            "patchManagement": 0,
            "edrSensor": 0,
            "networkAttackDefense": 0
        },
        "roles": {
               "relay": 1,
               "exchange": 0
        },
        "scanMode": {
           "type": 2,
           "computers": {
               "main": 1,
               "fallback": 2
           },
           "vms": {
               "main": 2
           }
        },
        "settings": {
           "scanBeforeInstall": false,
           "removeCompetitors": true,
           "customInstallationPath": "c:\\mypath\\bitdefender",
           "customGroupId": "5a4dff50b1a43ded0a7b23c8",
           "vmsCustomGroupId": "5a4dff50b1a43ded0a7b23c7"
        },
        "deploymentOptions": {
           "type": 1,
           "useCommunicationProxy": true,
           "proxyServer": "10.12.13.14",
           "proxyPort": 123,
           "proxyUsername": "user",
           "useCustomCommunicationServer": true,
           "communicationServer": "10.12.13.14"
        },
        "productType": 0
```

```
    }
}
```

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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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.

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

Return value

This method returns an Object containing a list of policy objects. The result 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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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.

Parameter	Type	Optional	Description
`type`	Number	No	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 • 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 • 30 - Endpoint Encryption Status • 31 - HyperDetect Activity • 32 - Network Patch Status • 33 - Sandbox Analyzer Failed Submissions

Parameter	Type	Optional	Description
			• 34 - Network Incidents
`targetIds`	Array	No	A list with the IDs of the targets for which to create the report. The target ID can be any 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 email addresses where to deliver the report.
`attachFilesAsArchive`	Boolean	Yes	The parameter defines whether the email should include an archive with the report files, or not.

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 – 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:

Name	Type	Description
		– 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: – 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. 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

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

Name	Type	Description
		– `True`, to include in the report endpoints with update disabled. – `False`, to exclude from the report endpoints with update disabled.
`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.

Name	Type	Description
		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 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 Status**

  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
`filterType`	integer	The member is mandatory. Possible values: – 0 - All endpoints – 1 - Only endpoints still infected
`detailedExport`	array	The member is optional. Possible values: – 1 - Include Endpoint Malware Status in PDF file

- **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.

Name	Type	Description
		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

- **HyperDetect 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

- **Network Patch Status**

  The object must contain these members:

Name	Type	Description
`filterType`	integer	The member is mandatory. Possible values: – 0 - All available patches – 1 - Only patches visible in Patch Inventory

- **Sandbox Analyzer Failed Submissions**

  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

- **Network Incidents**

  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

- **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**
- **Endpoint Encryption 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
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

occurrence	reportingInterval
	– 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 you to place the `{service}` name in the API URL. The allowed services are:

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

For example, 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 • 2 - Blocked Applications • 3 - Blocked Websites • 5 - Data Protection • 6 - Device Control Activity • 7 - Endpoint Modules Status • 8 - Endpoint Protection Status • 9 - Firewall 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 • 30 - Endpoint Encryption Status

Parameter	Type	Optional	Description
			• 31 - HyperDetect Activity • 32 - Network Patch Status • 33 - Sandbox Analyzer Failed Submissions • 34 - Network Incidents
`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 100 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 which is 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
}
```

2.6. Quarantine

The Quarantine API contains the following methods allowing the management of quarantine items.

- `getQuarantineItemsList` : retrieves the list of available quarantined items related to a company.
- `createRemoveQuarantineItemTask` : creates a task to remove quarantined items.
- `createEmptyQuarantineTask` : creates a task to empty the quarantined items list.
- `createRestoreQuarantineItemTask` : creates a task to restore quarantined items.
- `createRestoreQuarantineExchangeItemTask` : creates a task to restore exchange quarantined items.

API url: CONTROL_CENTER_APIs_ACCESS_URL/v1.0/jsonrpc/quarantine

2.6.1. getQuarantineItemsList

This method retrieves the list of quarantined items available for a company.

An item can be a file or an Microsoft Exchange object.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

- `computers`, for "Computers and Virtual Machines"
- `exchange`, for "Security for Exchange"

For example, the request URL for the `exchange` service is:

```
https://YOUR-HOSTNAME/api/v1.0/jsonrpc/quarantine/exchange
```

Parameters

Parameter	Type	Optional	Description
`endpointId`	String	Yes	The ID of the computer for which you want to retrieve the quarantined items. If not passed, he method returns the items quarantined in the entire network.
`page`	Number	Yes	The results page. The default value is 1.
`perPage`	Number	Yes	The number of items displayed in a page. The upper limit is 100 items per page. Default value is 30 items per page.
`filters`	Object	Yes	The filters to be used when querying the quarantine items list. For information regarding the available filters and how to use them, refer to "Available Filters" (p. 122).

Available Filters

You can use the `filters` parameter to query the quarantine by certain properties.

The query result is a list of quarantine items that match ALL selected filters.

These are the available filtering options:

Field	Type	Description
`threatName`	String	Filters the quarantined items by threat name. This filter is available for `computers` and `exchange` services.
`startDate`	String	Filters the items that were quarantined after the specified date. The format for `startDate` is in ISO 8601. This filter is available for `computers` and `exchange` services.
`endDate`	String	Filters the items that were quarantined before the specified date.

Field	Type	Description
		The format for `endDate` is in ISO 8601. This filter is available for `computers` and `exchange` services.
`filePath`	String	Filters the quarantined items by file path. This filter is available for `computers` service.
`ip`	String	Filters the quarantine items by IP address. This filter is available for `computers` service.
`actionStatus`	Integer	Filters the quarantine items by action status. The available values for `actionStatus` are: • `0` - None • `1` - Pending remove • `2` - Pending restore • `3` - Remove failed • `4` - Restore failed If the service is `exchange`, then the following will also be valid action statuses: • `16` - Pending Save • `17` - Failed Save This filter is available for `computers` and `exchange` services.

Important

- The fields `threatName`, `filePath` and `ip` work with partial matching.

  The filter returns the items which are exact match or start with the specified value. To use the specified value as a suffix, use the asterisk symbol (*).

  For example:

  If `filePath` is `C:\temp`, the API returns all items originating from this folder, including sub-folders.

If `filePath` is `*myfile.exe`, then the API returns a list of all myfile.exe files from anywhere on the system.

- The `Exchange` filters require a valid license key for Security for Exchange.

Return value

This method returns an Array containing objects with the quarantined items. Each entry in the array has the following structure:

- `page` - the current displayed page
- `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 quarantined items. Each entry in the list has the following fields:
  - `id`, the ID of the quarantined item,
  - `quarantinedOn`, the date and time when the object was quarantined,
  - `actionStatus`, the status of the action taken on the quarantined file: (`0` - None; `1` - Pending remove; `2` - Pending restore; `3` - Remove failed; `4` - Restore failed; `16` - Pending save; `17` - Failed save) ,
  - `endpointId`, the ID of the endpoint on which the threat was detected,
  - `endpointName`, the name of endpoint on which the threat was detected,
  - `endpointIP`, the IP of endpoint on which the threat was detected,
  - `canBeRestored`, has the value `True` if the restore operation is allowed, `False` otherwise,
  - `companyId`, the company ID,
  - `details`, more information related to the quarantined item. For information regarding the content of the details member, refer to "Contents of details" (p. 124).

Contents of details

For the `Computers and Virtual Machines` service, the `details` field has this structure:

Field name	Data type	Description
filePath	String	Path to the infected or suspicious file on the endpoint it wasdetected on

For Security for Exchange service, the details field has this structure:

Field name	Data type	Description
detectionPoint	Integer	The level where the detection took place. Possible values: • 0 - transport • 1 - mailbox • 2 - folder • 3 - on demand
itemType	Integer	The quarantined object type. Possible values: • 0 - attachement • 1 - email
threatStatus	String	The status of the object when scan is complete. The status shows if an email is spam or contains unwanted content, or if an attachment is malware infected, suspect of being infected, unwanted or unscannable. Possible values are: • 0 - spam • 1 - suspected • 2 - infected • 3 - attachement detection • 4 - content detection • 5 - unscannable
email	Object	• senderIP, a String containing the sender's IP address

Field name	Data type	Description
		• `senderEmail`, a String consisting in the sender's email address, as it appears in the email header **fieldFrom** • `subject`, a String with the subject of the quarantined email • `recipients`, an Array with the recipients, as they appear in the email header fields **To** and **Cc** • `realRecipients`, an Array containing the email addresses of the intended recipients

Example

Request :

```
{
    "params": {
        "endpointId": "5d36c255f23f730fa91944e2",
        "page": 2,
        "perPage": 1,
        "filters": {
            "threatName": "Virus 0",
            "actionStatus": 1,
            "startDate": "2019-07-28T11:31:28",
            "endDate": "2019-08-16T11:31:16",
            "filePath": "c:\\Virus0\\virus0.exe"
        }
    },
    "jsonrpc": "2.0",
    "method": "getQuarantineItemsList",
    "id": "5399c9b5-0b46-45e4-81aa-889952433d86"
}
```

Response :

```
This response example is for computers service:
   {
       "id": "5399c9b5-0b46-45e4-81aa-889952433d86",
       "jsonrpc": "2.0",
       "result": {
           "total": 2,
           "page": 2,
           "perPage": 1,
           "pagesCount": 2,
           "items": [
               {
                   "id": "5d3968e0f23f730ecb0f68c2",
                   "quarantinedOn": "2019-07-28T11:31:28",
                   "actionStatus": 1,
                   "endpointId": "5d36c255f23f730fa91944e2",
                   "endpointName": "Computer 1",
                   "endpointIP": "156.133.37.181",
                   "canBeRestored": false,
                   "canBeRemoved": false,
                   "threatName": "Virus 0",
                   "details": {
                       "filePath": "c:\\Virus0\\virus0.exe"
                   }
               }
           ]
       }
   }
This response example is for exchange service:
   {
       "id":"5399c9b5-0b46-45e4-81aa-889952433d86",
       "jsonrpc":"2.0",
       "result": {
            page: 2,
            pagesCount: 10,
            perPage: 1,
            total: 10
            items[{
                 "id": "5b7d219bb1a43d170b7b23ee",
                 "quarantinedOn": "2019-08-01T07:15:20",
                 "actionStatus": 1,
                 "endpointId": "5d36c255f23f730fa91944e2",
                 "endpointName": "Computer 1",
```

```
                    "endpointIP": "57.238.160.118",
                    "endpointAvailable": true,
                    "threatName": "Virus 0",
                    "details": {
                        "threatStatus": 4,
                        "itemType" : 0,
                        "detectionPoint": 1,
                        "email": {
                            "senderIP": "185.36.136.238",
                            "senderEmail": "test@test.com",
                            "subject":
                          "Test subject_5b7d2128b1a43da20c7b23c6",
                            "recipients": [
                               "receiver1@test.com", "
                               receiver2@test.com",
                            ]
                            "realRecipients": [
                               "receiver1@test.com", "
                               receiver2@test.com"
                          ]
                        }
                    }
                }]
            }
        }
```

2.6.2. createRemoveQuarantineItemTask

This method creates a new task to remove items from quarantine.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

- `exchange`, for "Security for Exchange"
- `computers`, for "Computers and Virtual Machines"

For example, the request URL for the `computers` service is:

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

Parameters

Parameter	Type	Optional	Description
quarantineItemsIds	Array	No	The list of quarantine items IDs

Return value

This method returns a Boolean which is True when the task was successfully created.

Example

Request :

```
{
      "params": {
          "quarantineItemsIds": [
              "63896b87b7894d0f367b23c6",
              "65896b87b7894d0f367b23c6"
          ]
      },
      "jsonrpc": "2.0",
      "method": "createRemoveQuarantineItemTask",
      "id": "5399c9b5-0b46-45e4-81aa-889952433d86"
}
```

Response :

```
{
     "id": "5399c9b5-0b46-45e4-81aa-889952433d86",
     "jsonrpc":"2.0",
     "result": True
}
```

2.6.3. createEmptyQuarantineTask

This method creates a new task to empty the quarantine.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

- `exchange`, for "Security for Exchange"
- `computers`, for "Computers and Virtual Machines"

For example, the request URL for the `computers` service is:

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

Parameters

No input parameters are required.

Return value

This method returns a Boolean which is True when the task was successfully created.

Example

Request :

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

Response :

```
{
    "id": "5399c9b5-0b46-45e4-81aa-889952433d86",
    "jsonrpc":"2.0",
    "result": True
}
```

2.6.4. createRestoreQuarantineItemTask

This method creates a new task to restore items from the quarantine.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

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

For example, the request URL for the `computers` service is:

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

Parameters

Parameter	Type	Optional	Description
`quarantineItemsIds`	Array	No	The list of IDs for the quarantined items. You can restore maximum 100 items once.
`locationToRestore`	String	Yes	The absolute path to the folder where the items will be restored. If the parameter is not set, the original location will be used.
`addExclusionInPolicy`	Boolean	Yes	Exclude the files to be restored from future scans. Exclusions do not apply to items with the **Default Policy** assigned. The default value for this parameter is `False`.

Return value

This method returns a Boolean which is True when the task was successfully created.

Example

Request :

```
{
      "params": {
          "quarantineItemsIds": [
              "63896b87b7894d0f367b23c6",
              "65896b87b7894d0f367b23c6"
           ],
          "locationToRestore": "C:\RestoreDirectory"
          "addExclusionInPolicy": true
      },
      "jsonrpc": "2.0",
      "method": "createRestoreQuarantineItemTask",
      "id": "5399c9b5-0b46-45e4-81aa-889952433d86"
}
```

Response :

```
{
    "id": "5399c9b5-0b46-45e4-81aa-889952433d86",
    "jsonrpc":"2.0",
    "result": True
}
```

2.6.5. createRestoreQuarantineExchangeItemTask

This method creates a new task to restore items from the quarantine for Exchange Servers.

Services

This method requires you to place the `{service}` name in the API URL. The allowed services are:

- `exchange`, for "Security for Exchange"

For example, the request URL for the `exchange` service is:

`https://YOUR-HOSTNAME/api/v1.0/jsonrpc/quarantine/exchange`

Parameters

Parameter	Type	Optional	Description
`quarantineItemsIds`	Array	No	The list of IDs for the quarantined items. You can restore maximum 100 items once.
`username`	String	No	The username of an Microsoft Exchange user. The username must include the domain name.
`password`	String	No	The password of an Exchange user
`email`	String	Yes	The email address of the Exchange user. This parameter is necessary when the email address is different from the username.
`ewsUrl`	String	Yes	The Exchange Web Services URL .The EWS URL is necessary when the Exchange Autodiscovery does not work.

Return value

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

Example

Request :

```
{
      "params": {
          "quarantineItemsIds": [
              "63896b87b7894d0f367b23c6",
              "65896b87b7894d0f367b23c6"
          ],
          "username": "user@domain",
          "password": "userPassword"
      },
      "jsonrpc": "2.0",
      "method": "createRestoreQuarantineExchangeItemTask",
```

```
        "id": "5399c9b5-0b46-45e4-81aa-889952433d86"
}
```

Response :

```
{
    "id": "5399c9b5-0b46-45e4-81aa-889952433d86",
    "jsonrpc":"2.0",
    "result": True
}
```

2.7. General

The General API includes methods for general use without the need to enable a specific API to call any of these methods.

- `getApiKeyDetails` : returns details about the API key used.

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

2.7.1. getApiKeyDetails

This method returns details about the API key used.

Parameters

No input parameters are required.

Return value

This method returns an Object containing the details of the API key:

- `enabledApis` - an Array containing the list of enabled APIs
- `createdAt` - a String representing the UTC date and time when the API key was generated

Example

Request :

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

Response :

```
{
    "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
    "jsonrpc":"2.0",
    "result": {
         "enabledApis": [
              "packages",
              "network",
              "policies",
              "reports"
          ],
         "createdAt": "2019-04-11T12:00:54"
   }
}
```

2.8. Sandbox

Sandbox Analyzer API retrieves metadata related to the Sandbox Analyzer instances, images and submissions.

- `getSandboxAnalyzerInstancesList` : lists Sandbox Analyzer instances.
- `getImagesList` : lists images for a Sandbox Analyzer instance.
- `getSubmissionStatus` : returns the status of a submission.
- `getDetonationDetails` : returns the details of a submission.

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

2.8.1. getSandboxAnalyzerInstancesList

This method lists the Sandbox Analyzer instances in the **Infrastructure** menu.

Parameters

Parameter	Type	Optional	Description
`page`	Number	Yes	The results page. Default value: `1`.
`perPage`	Number	Yes	The number of items displayed per page. The upper limit is 100 items per page. Default value: `30`.

Return value

This method returns an Object containing information regarding the Sandbox Analyzer instances. The object has the following structure:

- `page` - the current page displayed
- `pagesCount` - the total number of available pages
- `perPage` - the number of returned items per page
- `items` - the list of Sandbox Analyzer instances. Each item has the following fields:
  - `sandboxId`, the ID of the Sandbox Analyzer instance.
  - `name`, the name of the Sandbox Analyzer instance.
  - `ip`, the IP address of the Sandbox Analyzer instance.
  - `macs`, the MAC addresses of the Sandbox Analyzer instance.
  - `ssid`, the Active Directory SID of the Sandbox Analyzer instance.
  - `detonatedSamples`, the overall number of samples analyzed by the Sandbox Analyzer instance.
  - `diskUsage`, the percentage of the disk space that Sandbox Analyzer occupies in the datastore.
  - `installationStatus`, the status of the Sandbox Analyzer installation process. It can have one of the following values:
    - `0` - Not installed
    - `1` - Installed
    - `2` - Installing
    - `3` - Installation failed
  - `lastSeen`, the date of the last synchronization with Control Center.

  - `configuredConcurrentDetonations`, the number of virtual machines allocated to detonate samples.
  - `maximumConcurrentDetonations`, the maximum number of virtual machines that the Sandbox Analyzer instance can create to detonate samples.
  - `submissionUrl`, the URL for submitting files for analysis.
- `total` - the total number of items

Example

Request :

```
{
   "method": "getSandboxAnalyzerInstancesList",
   "params": {
      "page": 1,
      "perPage": 20
   },
   "jsonrpc": "2.0",
   "id": "91d6430d-bfd4-494f-8d4d-4947406d21a7"
}
```

Response :

```
{
   "id": "91d6430d-bfd4-494f-8d4d-4947406d21a7",
   "jsonrpc": "2.0",
   "result": {
      "page": 1,
      "pagesCount": 1,
      "perPage": 20,
      "total": 1,
      "items": [
         {
             "sandboxId": "5c419e6e26df3d367c49de18",
             "name": "sandbox1",
             "ip": "10.10.20.1",
             "macs": [
                 "00-14-22-01-23-45"
```

```
                ],
                "ssid": "",
                "detonatedSamples": 0,
                "diskUsage": 250,
                "installationStatus": 1,
                "lastSeen": "2019-01-18T11:37:50",
                "configuredConcurrentDetonations": 0,
                "maximumConcurrentDetonations": 10,
                "submissionUrl":
                    "https://10.10.20.1:443/api/v1/upload"
            }
        ]
    }
}
```


2.8.2. getImagesList

This method lists all images available on a Sandbox Analyzer instance.

Parameters

Parameter	Type	Optional	Description
`sandboxId`	String	No	The ID of the Sandbox Analyzer instance for which the images list will be returned.
`page`	Number	Yes	The results page number. Default value: `1`.
`perPage`	Number	Yes	The number of items displayed in a page. The upper limit is 100 items per page. Default value: `30`.

Return value

This method returns an Object containing information regarding the images. The object has the following structure:

- `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 images. Each item in the list has the following fields:
  - `id`, the ID of the image.
  - `name`, the name of the image.
  - `status`, the status of the image. It can have one of the following values:
    - `1` - New
    - `2` - Failed
    - `3` - Ready
  - `operatingSystem`, the operating system of the image.
  - `dateAdded`, the date on which the image was added.
  - `isDefault`, a Boolean which has the value `True` when the image is set as default. `False` otherwise.
  - `actionInProgress`, a Boolean which has the value `True` when there is an action in progress for this image.
- `total` - the total number of items

Example

Request :

```
{
   "method": "getImagesList",
   "params": {
      "sandboxId": "5c419e6e26df3d367c49de18",
      "page": 1,
      "perPage": 20
   },
   "jsonrpc": "2.0",
   "id": "91d6430d-bfd4-494f-8d4d-4947406d21a7"
}
```

Response :

```
{
   "id": "91d6430d-bfd4-494f-8d4d-4947406d21a7",
   "jsonrpc": "2.0",
```

```
    "result": {
       "page": 1,
       "pagesCount": 1,
       "perPage": 20,
       "total": 1,
       "items": [
          {
              "id": "924cca0d49cc7e350a44502b0bb9026a",
              "name": "image1",
              "status": 1,
              "operatingSystem": "Windows 10",
              "dateAdded": "2019-01-18T09:20:50",
              "isDefault": true,
              "actionInProgress": false
          }
       ]
    }
}
```

2.8.3. getSubmissionStatus

Returns the final status of the detonation.

Parameters

Parameter	Type	Optional	Description
`submissionId`	String	No	The ID of the submission for which the status should be retrieved.

Return value

This method returns an Object containing the status:

- `status` - an Integer representing the final status. It can have one of the following values:
  - `1` - completed, if the detonation was successful
  - `2` - pending, if the detonation is currently in progress
  - `3` - failed, if the detonation failed

- `4` - not supported, if the file cannot be detonated

Example

Request :

```
{
   "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
   "jsonrpc": "2.0",
   "method": "getSubmissionStatus",
   "params": {
      "submissionId": "sp02_1547807011_936_e5"
   }
}
```

Response :

```
{
  "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
  "jsonrpc": "2.0",
  "result": {
    "status": 1
  }
}
```

2.8.4. getDetonationDetails

The method returns the details of a submission, including a URL pointing to the HTML report.

Parameters

Parameter	Type	Optional	Description
`submissionId`	String	No	The ID of the submission for which the detonation details should be retrieved.

Return value

This method returns an Object containing the details of a completed detonation. The object has the following structure:

- `detailsReportUrl` - a String containing the URL from where the HTML report is available for download.
- `score` - an Integer in the range 0-100 representing the severity of the threat, if any.
- `verdict` - an Integer having one of the following values:
  - 0, if clean.
  - 1, if infected.
  - 2, if unsupported.
- `mitreTags` - an Array of Objects with the following structure:
  - `category` a String holding the MITRE category.
  - `techniques` an Array of Strings holding the MITRE techniques.

Example

Request :

```
{
   "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
   "jsonrpc": "2.0",
   "method": "getDetonationDetails",
   "params": {
      "submissionId": "sp02_1547807011_936_e5"
   }
}
```

Response :

```
{
   "id": "787b5e36-89a8-4353-88b9-6b7a32e9c87f",
   "jsonrpc": "2.0",
   "result": {
      "detailsReportUrl":
          "https://10.10.20.1:443/api/v1/report?report_id=asd",
```

```
        "score": 30,
        "verdict": 0,
        "mitreTags": [
            {
                "category": "Defense Evasion",
                "techniques": [
                    "Modify Registry",
                ]
            },
            {
                "category": "Persistence",
                "techniques": [
                    ".bash_profile and .bashrc",
                    "Account Manipulation",
                ]
            },
        ]
    }
}
```


2.9. Sandbox Portal

The Sandbox Portal API exposes functionality for working directly with the Bitdefender Sandbox Analyzer. While the other GravityZone Sandbox APIs expose metadata for the detonations, the purpose of the Sandbox Portal API is to submit samples and to download the detailed report of the analysis.

The authorization is made using the same API key used for the GravityZone Sandbox APIs.

- `sample submission`: submit sample for detonation.
- `report`: retrieve the detailed report for a detonation.

API URL: https://SANDBOX_IP/api/v1/

Note
The **Sandbox Analyzer Portal API** is hosted on the**Sandbox Analyzer virtual appliance**.

2.9.1. Sample Submission

This API endpoint makes a submission to Sandbox Analyzer.

Submission URL: https://SANDBOX_IP/api/v1/upload

The API endpoint expects a HTTP multipart request, which contains a JSON with detonation options and a binary file. When making a submission via URL, a binary file should not be included in the request.

Options in JSON format

Field	Type	Optional	Description
`imageId`	String	No	The ID of the image which will be used to detonate the sample.
`detonation`	Object	No	The Object has the following structure: • `type` - a String representing the type of information submitted. It may have one of the following values: `file` - when submitting a binary file, or `url` - when submitting a URL. • `detonationProfile` - a String containing the detonation profile. This allows you to choose between sandbox detonation throughput and detection accuracy, or to balance them. Possible values are: – `low` - increased Sandbox Analyzer throughput with reduced detonation analysis complexity. The accuracy of the detection remains in acceptable standards. – `medium` - the optimal balance between detonation time and analysis accuracy. – `high` - the best analysis accuracy. The side effect is a less than optimal detonation throughput.

Field	Type	Optional	Description
			• `url` - a String containing the URL to be analyzed. This attribute is optional. • `fileName` - a String representing the name of the file to be shown in console. This attribute is optional. If omitted, Sandbox Analyzer generates one. • `archivePassword` - a String containing the decryption password. This attribute is optional. If omitted, Sandbox Analyzer will not be able to analyze the contents of an encrypted archive.
`detonationOptions`	Object	Yes	An Object containing detonation options. You can find the options described in the next section.

Detonation options

All options are optional. If an option is omitted, Sandbox Analyzer uses the default value.

Option	Type	Default	Description
`commandLineArguments`	String	No arguments provided	The list of command line arguments that Sandbox Analyzer uses when detonating the sample. This option is available only for file submissions.
`timeLimit`	Number	6 minutes	The maximum number of minutes that a detonation can last.

Option	Type	Default	Description
`numberOfReruns`	Number	2 reruns	The number of detonation attempts, in case of failure.
`preFiltering`	Boolean	True	Specifies whether Sandbox Analyzer caches previously analyzed samples (`True`) or not.
`internetAccess`	Boolean	True	Sets the internet access of the VM. If `True`, the VM can access the internet.

For example, the JSON for a sample with an encrypted archive, to be detonated with command lines arguments, on a VM without internet access, should look like this:

```
{
    "imageId": "1787b5e3689a8435388b96b7a32e9c87f",
    "detonation": {
        "type": "file",
        "detonationProfile": "medium",
        "fileName: "infected.zip",
        "archivePassword": "123infected"
    },
    "detonationOptions": {
        "commandLineArguments": "--extraParam 41",
        "internetAccess": false
    }
}
```

Next sample is a JSON for a URL submission without options.

```
{
    "imageId": "1787b5e3689a8435388b96b7a32e9c87f",
    "detonation": {
        "type": "url",
        "detonationProfile": "medium",
        "url": "http://storage.infected.info/images/test.php"
    }
}
```

Return value

This method returns an Object containing information regarding the submission. It has the following structure:

- `code` - an Integer representing the HTTP status code
- `message` - a String with the response description
- `submissionId` - the ID allocated to the submitted sample. Omitted in case of error.
- `errors` - an Array of Strings in case of bad request. This attribute is returned only in case of error.

Example

Given that Sandbox Portal API for making submissions is not JSON RPC, but rather HTTP multipart request, we'll also provide a couple of cURL examples.

The following examples show how the authorization header is set using the API key: `UjlMS+0m1l9IUZjpjWyJG8gbnv2Mta4T`.

Read detonation options from file:

```
curl -X POST \
    https://{sandbox_ip}:9090/api/v1/upload \
    -H 'Authorization: Basic \
        VWpsTVMrMG0xbDlJVVpqcGpXeUpHOGdibnYyTXRhNFQ=' \
    -F 'options=@{/path/to/json_with_detonation_options}' \
    -F 'upload_file=@{/path/to/binary_file}'
```

Note
The first part of the request has to be a JSON containing detonation options, otherwise the submission will fail.
The name of the multipart field associated with the submitted file has to be `upload_file`, otherwise the submission will fail.

Generate detonation options on-the-fly:

```
curl -X POST \
    https://{sandbox_ip}:9090/api/v1/upload \
    -H 'Authorization: Basic \
```

```
        VWpsTVMrMG0xbDlJVVpqcGpXeUpHOGdibnYyTXRhNFQ=' \
    -F 'options={
            "imageId": "1787b5e3689a8435388b96b7a32e9c87f",
            "detonation": {
                "type": "url",
                "detonationProfile": "medium",
                "url": "http://www.infected.info/test.php"
            }
        }'
```

Response:

```
{
    "code": 200,
    "message": "Ok",
    "submissionId": "sp02_1547807011_936_e5",
    "file": "infected.zip"
}
```

2.9.2. Report

This API endpoint retrieves an HTML report from Sandbox Analyzer.

Report URL: https://SANDBOX_IP/api/v1/report?report_id=REPORT_ID

The endpoint returns a deflatable GZIP archive containing the HTML file in case of success (HTTP status code 200), or a JSON in case of error.

Note
`report_id` is passed as a GET variable to the API endpoint.

2.9.3. Error Handling

In case of error, the Sandbox Portal replies with a relevant HTTP status code. The list of possible status codes is, but not limited to:

- `400` Bad Request
- `401` Unauthorized
- `402` Malformed Request
- `403` Forbidden

- `404` Not Found
- `405` Method Not Allowed
- `429` Too Many Requests
- `500` Internal Server Error

In addition to the HTTP status code, the body of the response contains a JSON describing the error.

The JSON has the following fields:

- `code` - an Integer representing the HTTP status code
- `message` - a String with the description of the error
- `errors` - an Array of Strings which provide additional information about the error. This field is optional.

Example of a response when trying to submit a URL with invalid options:

```
{
    "code": 400,
    "message": "Failed validating request",
    "errors": [
        "Invalid URL provided",
        "Invalid detonation object",
        "Invalid detonation options object"
    ]
}
```

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 list of available packages.

```
import base64
import requests
# Generate Authorization header from API key
apiKey = "UjlMS+0m1l9IUZjpjWyJG8gbnv2Mta4T"
auth = base64.b64encode((apiKey + ":").encode("UTF-8"))\
    .decode("UTF-8")
authorizationHeader = "Basic " + auth
json = {
    "method": "getPackagesList",
    "params": {},
    "jsonrpc": "2.0",
    "id": 123
}
result = requests.post(
"https://{domain}/api/v1.0/jsonrpc/packages",
json=json,
verify=False,
headers = {
"Content-Type": "application/json",
"Authorization": authorizationHeader
}).json()

print(result)

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}
//  }
```

3.5. PowerShell Example

This is an example PowerShell script. It provides the basics to make an API call to a GravityZone API endpoint.

Note
We wrote the operations in this script explicitly for didactic purposes. Feel free to optimize them for your practical use cases, should you feel it necessary.

```
# Store the API token (change this to your API key)
# For details, refer to the "API Keys" section of this guide.

$api_key = 'UjlMS+0m1l9IUZjpjWyJG8gbnv2Mta4T'


# build the login string (pass is an empty string)

$user = $api_key
$pass = ""
$login = $user + ":" + $pass


# encode the login string to base64

$bytes= [System.Text.Encoding]::UTF8.GetBytes($login)
$encodedlogin=[Convert]::ToBase64String($bytes)


# prepend "Basic " to the encoded login string to obtain
# the auth header
```

```
$authheader = "Basic " + $encodedlogin


# Replace the base_uri string with the correct one
# for your console

$base_uri = "https://cloud.gravityzone.bitdefender.com/api"


# Replace the api_endpoint string with the correct one for
# the method used in the request_data
# For details, defer to the "API Requests" section
# of this guide.

$api_endpoint = "/v1.0/jsonrpc/network"


# Build the request URI

$request_uri = $base_uri + $api_endpoint


# Store the request body in a JSON variable.
# Define the API call method and its parameteres.
# For more details on each API method, refer to the "Reference"
# chapter of this guide.

$request_data = '{
 "id":"123456789",
 "jsonrpc":"2.0",
 "method":"getEndpointsList",
 "params":
  {
   "page":1,
   "perPage":3
  }
}'


# All required resources are now set.

# You have two options to make the API call.
```

```
# First option:
# Add all call parameters in one structure, then call
# Invoke-RestMethod with it.

$params = @{
    Uri         = $request_uri
    Headers     = @{
  'Authorization' = "$authheader"
  'Content-Type' = "application/json"
 }
    Method      = 'POST'
    Body        = $request_data
    ContentType = 'application/json'
}

$response = Invoke-RestMethod @params


# Second option:
# Build the headers structure, but specify the
# Invoke-RestMethod parameters inline.

$headers = New-Object `
"System.Collections.Generic.Dictionary[[String],[String]]"
$headers.Add("Authorization",$authheader)
$headers.Add("Content-Type","application/json")

$response2 = Invoke-RestMethod -Uri $request_uri `
-Headers $headers -ContentType 'application/json' `
-Method Post -Body $request_data


# Random examples of how to address/display the obtained
# call results from the $response and $response2 variables

Write-Output '$response'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response
Write-Output '$response |ConvertTo-Json'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response |ConvertTo-Json
Write-Output '$response.result | ConvertTo-Json'
```

```
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response.result | ConvertTo-Json
Write-Output '$response.result.items |ConvertTo-Json'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response.result.items |ConvertTo-Json

Write-Output '$response2'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response2
Write-Output '$response2.result'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response2.result
Write-Output '$response2.result.items'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response2.result.items
Write-Output '$response2.result.items.id[0]'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response2.result.items.id[0]
Write-Output '$response2.result.items.name[1]'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response2.result.items.name[1]
Write-Output '$response2.result.items[1] |ConvertTo-Json'
Write-Output "~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
$response2.result.items[1] |ConvertTo-Json
```

3.6. VBScript Example

This is a VBScript example. It provides the basics to make an API call to a GravityZone API endpoint.

Note
We wrote the operations in this script explicitly for didactic purposes. Feel free to optimize them for your practical use cases, should you feel it necessary.

```
'These are for displaying the results of the call.

Set fso = CreateObject ("Scripting.FileSystemObject")
Set stdout = fso.GetStandardStream (1)
Set stderr = fso.GetStandardStream (2)
```

```
'These are some helping funtions used for base64 encoding
'of the authorization header.

Private Function Stream_StringToBinary(Text)
    Const adTypeText = 2
    Const adTypeBinary = 1
    Dim BinaryStream 'As New Stream
    Set BinaryStream = CreateObject("ADODB.Stream")
    BinaryStream.Type = adTypeText
    BinaryStream.CharSet = "us-ascii"
    BinaryStream.Open
    BinaryStream.WriteText Text
    BinaryStream.Position = 0
    BinaryStream.Type = adTypeBinary
    BinaryStream.Position = 0
    Stream_StringToBinary = BinaryStream.Read
    Set BinaryStream = Nothing
End Function

Function Base64Encode(sText)
    Dim oXML, oNode
    Set oXML = CreateObject("Msxml2.DOMDocument.3.0")
    Set oNode = oXML.CreateElement("base64")
    oNode.dataType = "bin.base64"
    oNode.nodeTypedValue = Stream_StringToBinary(sText)
    Base64Encode = Replace(oNode.text, chr(10), "")
    Set oNode = Nothing
    Set oXML = Nothing
End Function


'Store the API token.
'Make sure to change the string with your actual API key.
'For more information, refer to the "API Keys" section
'of this guide.

api_key = "UjlMS+0m1l9IUZjpjWyJG8gbnv2Mta4T"


'Build the login string.
```

```
'Note: pass is an empty string.

user = api_key
pass = ""
login = user & ":" & pass


'Encode the login string to base64.

encodedlogin = Base64Encode(login)


'Prepend "Basic " to the encoded login string to obtain
'the auth header.

authheader = "Basic " & encodedlogin


'Change the base_uri string with the correct one for your console.

base_uri = "https://cloud.gravityzone.bitdefender.com/api"


'Change the api_endpoint string with the correct one
'for the method used in the request_data.
'For details, refer to "API Requests" section of this guide.

api_endpoint = "/v1.0/jsonrpc/network"


'Build the request URI.

request_uri = base_uri & api_endpoint


'Create the body of the request.
'Define the API call method and its parameteres.
'For more information, refer to the "Reference" chapter
'of this guide.
'Note: Due to limited page width, the strJSONRequest string
'is on multiple lines. You need to put it on one line.
```

```
strJSONRequest = "{""id"":""123456789"",
 ""jsonrpc"":""2.0"",
 ""method"":""getEndpointsList"",
 ""params"":{""page"":1,""perPage"":3}}"


'All required resources are set.

'Make the API call.

Set objHTTP = CreateObject("MSXML2.ServerXMLHTTP")
objHTTP.Open "POST", request_uri, False
objHTTP.setRequestHeader "Authorization", authheader
objHTTP.setRequestHeader "Content-Type", "application/json"
objHTTP.send (strJSONRequest)


'Examples of how to display call reponse:

stdout.WriteLine "Response Code: " & objHTTP.status
stdout.WriteLine "Response Headers: " & objHTTP.getAllResponseHeaders
stdout.WriteLine "Response Data: " & objHTTP.ResponseText
```

A. Appendices

A.1. API Error Codes

Code	Description
4030001	This error is thrown when operation is not permitted, because the feature is not available on this platform
4030003	This error is thrown on MOVE ENDPOINTS validation process because Only MSP users can move endpoints to other companies
4050001	This error is thrown on MOVE ENDPOINTS validation process when destination group is invalid
4050002	This error is thrown on MOVE ENDPOINTS validation process when destination group Id param is not a string
4050003	This error is thrown on MOVE ENDPOINTS validation process when endpointIds param is not a non-empty array
4050004	This error is thrown on MOVE ENDPOINTS validation process when target endpoint is unmanaged
4050005	This error is thrown on MOVE ENDPOINTS validation process when target endpoint is under same company as destination
4050006	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because it is not movable
4050007	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because it does not have unified client app id
4050008	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because it already has a Move task in progress
4050009	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because it has been already moved from this company
4050010	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because source company is not directly under current user's company

Code	Description
4050011	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because of MA issues
4050012	This error is thrown on MOVE ENDPOINTS validation process because target endpoint cannot be moved between companies with different BEST customizations
4050013	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because it has encrypted volumes
4050014	This error is thrown on MOVE ENDPOINTS validation process when target endpoint cannot be moved because source company doesn't have monthly license
4050015	This error is thrown on MOVE ENDPOINTS validation process when destination company license is not Monthly License or destination company is not a direct company
4050016	This error is thrown on MOVE ENDPOINTS validation process when the target endpoint cannot be moved because the source company has paid subscription and the destination company has trial subscription
4050017	This error is thrown on DELETE CUSTOM CONTAINER GROUP validation process when the target groupId cannot be removed because contains container hosts
4050018	This error is thrown on MOVE CUSTOM GROUP validation process when trying to move entity into Containers from outside source
4050019	This error is thrown on MOVE CUSTOM CONTAINER GROUP validation process when trying to move entity from Containers to outside source
4050020	This error is thrown on DELETE CUSTOM CONTAINER GROUP validation process when trying to delete a container host folder
4050021	This error is thrown on DELETE CUSTOM CONTAINER GROUP validation process when trying to delete a container
4050022	This error is thrown on MOVE CUSTOM GROUP validation process when trying to move container host while is synchronizing