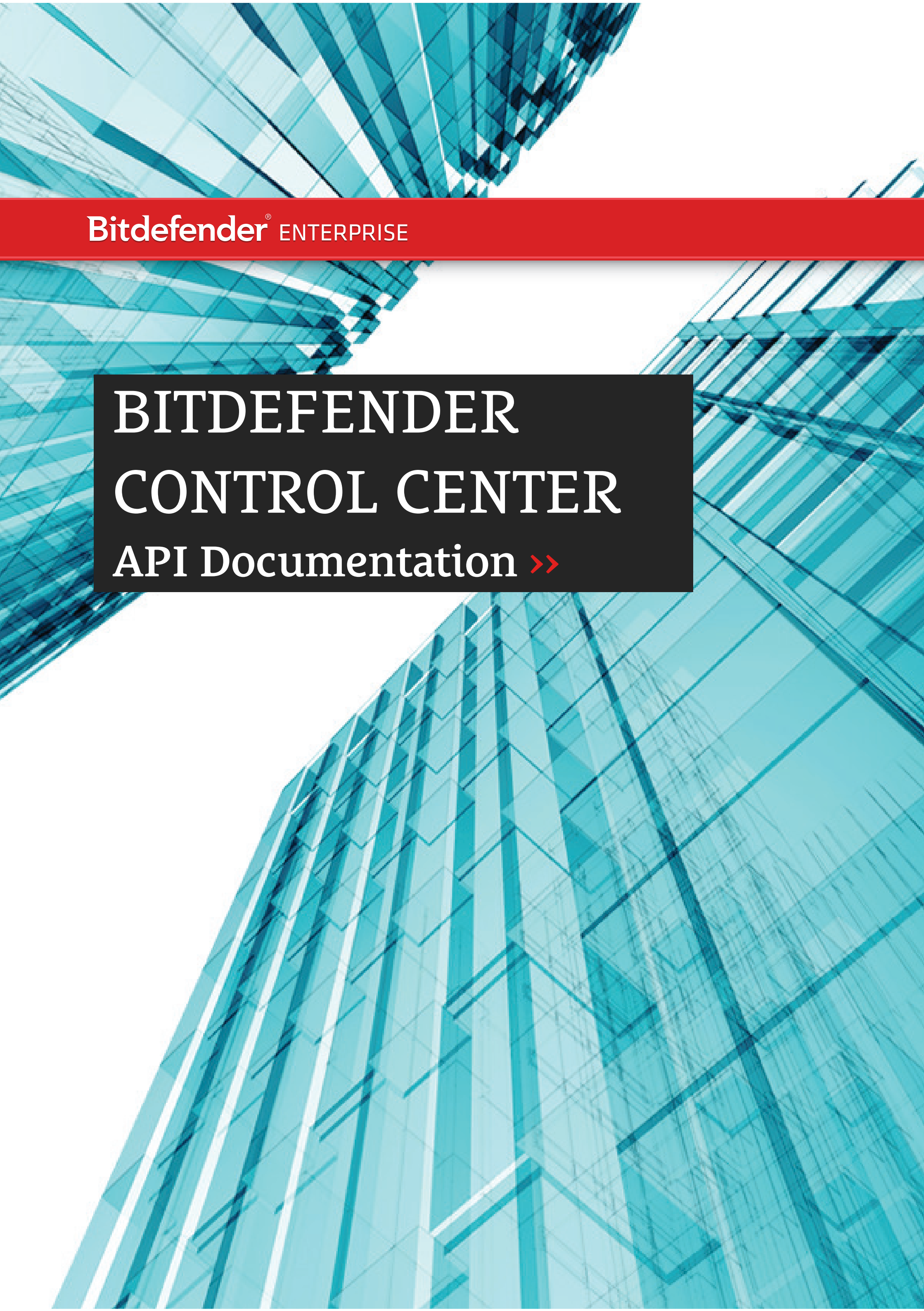


Bitdefender® ENTERPRISE

BITDEFENDER CONTROL CENTER

API Documentation »

Bitdefender Control Center API Documentation

Publication date 2015.01.19

Legal Notice

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

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

Bitdefender®

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.

Here is an example of API call creating a company inside Bitdefender Control Center:

```
{
    "id": "91d6430d-bfd4-494f-8d4d-4947406d21a7",
    "jsonrpc": "2.0",
    "method": "createCompany",
    "params": {
        "type": 0,
        "name": "My Company"
    }
}
```

For this call, the following response is sent back to the application:

```
{
    "id":"91d6430d-bfd4-494f-8d4d-4947406d21a7",
    "jsonrpc":"2.0",
    "result": "5493dcd2b1a43df00b7b23c6"
}
```

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.

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 to have the `Content-Type` header set to `application/json`.

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

Bitdefender Control Center exposes multiple API endpoints targeting a distinct area in the product. Each API endpoint exposes a set of methods related to a designated product area. The base URL for all API endpoints is: https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/. The full URL of the API endpoint is obtained by adding the API name to the base URL.

Currently, the following APIs are being exposed:

1. **Companies**. The endpoint is:

   https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/companies.
2. **Licensing**. The endpoint is:

   https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/licensing.
3. **Network**. The endpoint is:

   https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/network.
4. **Packages**. The endpoint is:

   https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/packages.

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

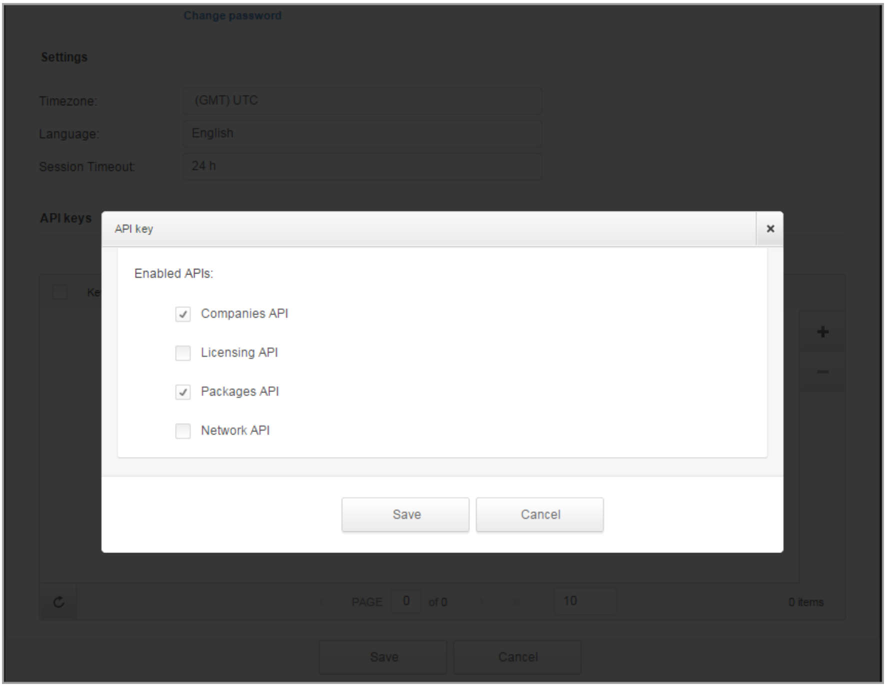
Note
Batch requests and notifications are not currently supported by Bitdefender Control Center.

1.2. 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 API endpoints. The allowed API endpoints are selected at the time the API key is generated.

To generate API keys:

1. Log in to https://gravityzone.bitdefender.com/ using your partner account.
2. Point to 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 right side of the table.
4. Select the APIs that you want to use.



Change password

Settings

Timezone: (GMT) UTC

Language: English

Session Timeout: 24 h

API keys

API key

Enabled APIs:

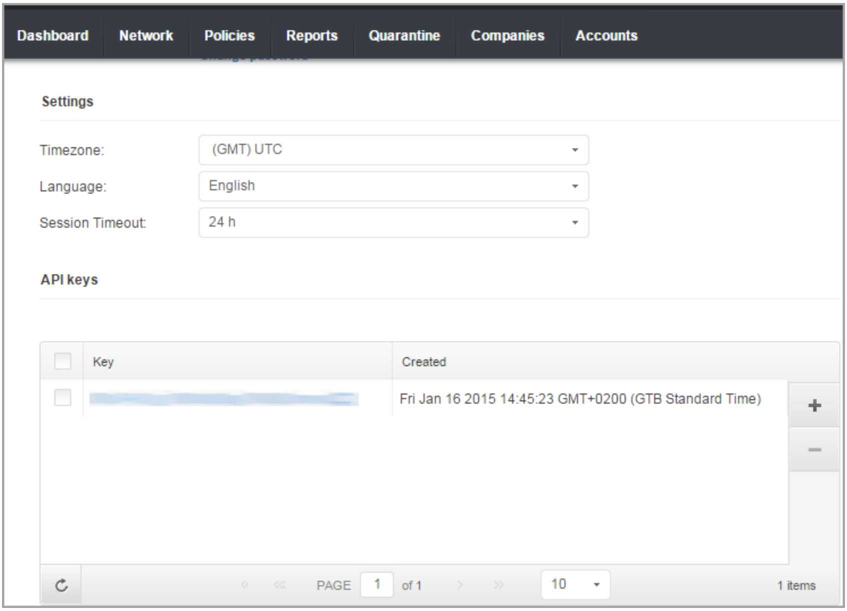
☑ Companies API

□ Licensing API

☑ Packages API

□ Network API

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



Settings

Timezone: (GMT) UTC

Language: English

Session Timeout: 24 h

API keys

□	Key	Created
□	[redacted]	Fri Jan 16 2015 14:45:23 GMT+0200 (GTB Standard Time)

Important

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

Bitdefender Control Center also allows partners to delete the previously generated API keys if these are no longer needed.

1.3. 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 a API endpoint.

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`.

1.4. 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 for errors:

- 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 endpoint is not enabled for 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 failed due to server errors (e.g. a required parameter is not passed).

2. Reference

2.1. Companies

The Companies API includes several methods allowing the management of companies:

- `createCompany` : adds a new company.
- `updateCompanyDetails` : updates the details of a company, such as name or type.
- `getCompanyDetails` : retrieves the details of a company.
- `suspendCompany` : disables access to Control Center for all user accounts of a company.
- `activateCompany` : activates a previously suspended company.

API endpoint: https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/companies

2.1.1. createCompany

This method creates a customer or partner company account.

Parameters

Parameter	Type	Optional	Description
`type`	Number	No	The company type. Can have the following values: 0 for Partner companies, 1 for Customer companies.
`name`	String	No	The company name. Must be unique.
`address`	String	Yes	The company address.
`phone`	String	Yes	The company phone number.
`canBeManagedByAbove`	Boolean	Yes	An option defining if the security of the new company can be managed by other companies. Available values: `true` or `false`. The default value is `true`.
`accountEmail`	String	Yes	The email for the new user account which will be created and linked to the new company. If the parameter `canBeManagedByAbove` is set to `false`,

Parameter	Type	Optional	Description
			the `accountEmail` parameter must be passed.
`accountFullName`	String	Yes	The full name of the new user account which will be created and linked to the new company. If the parameter `canBeManagedByAbove` is set to `false`, the `accountFullName` parameter must be passed.
`accountTimezone`	String	Yes	The timezone of the new user account which will be created and linked to the new company. The default value is `Europe/Bucharest`.
`accountLanguage`	String	Yes	The user interface language for the new user account which will be created and linked to the new company. The default value is `en_US`.

Return value

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

Example

Request :

```
{
     "params": {
        "type": 1,
        "name": "Customer LTD",
        "address": "Str Example No 1",
        "phone": "0040740000000",
        "canBeManagedByAbove": true,
        "accountEmail": "customer@example.com",
        "accountFullName": "Customer account"
     },
     "jsonrpc": "2.0",
     "method": "createCompany",
     "id": "e249c22c-0ada-4772-a9f1-ee1cbb322588"
}
```

Response :

```
{
    "id":"e249c22c-0ada-4772-a9f1-ee1cbb322588",
    "jsonrpc":"2.0",
    "result": "5493dcd2b1a43df00b7b23c6"
}
```

2.1.2. updateCompanyDetails

This method updates the details of a partner or customer company.

Parameters

Parameter	Type	Optional	Description
`id`	String	No	The id of the company to be updated.
`type`	Number	Yes	The company type. Available values: 0 for Partner companies, 1 for Customer companies. If not set, the company type will not be changed.
`name`	String	Yes	The company's name. It must be unique. If not set, the company's name will not be changed.
`address`	String	Yes	The company's address. If not set, the company's address will not be changed.
`phone`	String	Yes	The company's phone number. If not set, the company's phone number will not be changed.

Return value

This method does not return any value.

Example

Request :

```
{
     "params": {
        "id" : "5493dcd2b1a43df00b7b23c6",
        "type": 0,
        "name": "Customer to Partner LTD",
        "address": "Str Example No 1",
        "phone": "0040740000001"
     },
     "jsonrpc": "2.0",
     "method": "updateCompanyDetails",
     "id": "60357f0e-94da-463c-ba36-f50f2ef8c34f"
}
```

Response :

```
   {
       "id":"60357f0e-94da-463c-ba36-f50f2ef8c34f",
       "jsonrpc":"2.0",
       "result": null
   }
```

2.1.3. getCompanyDetails

This method retrieves the information contained in the details of a company.

Parameters

Parameter	Type	Optional	Description
`companyId`	String	Yes	The company's ID. The default value is the ID of the root company.

Return value

This method returns an Object containing the details of the selected company:

- `type` - company type: 0 for Partner, 1 for Customer
- `name` - the name of the company
- `id` - id of the company
- `address` - the address of the company
- `phone` - the phone of the company
- `canBeManagedByAbove` - `true` if the security for this company can be managed by other companies, `false` otherwise,
- `isSuspended` - `true` if the company is suspended, `false` otherwise

Example

Request :

```
{
     "params": {
        "companyId" : "5493dcd2b1a43df00b7b23c6"
     },
     "jsonrpc": "2.0",
     "method": "getCompanyDetails",
     "id": "0df7568c-59c1-48e0-a31b-18d83e6d9810"
}
```

Response :

```
{
    "id":"0df7568c-59c1-48e0-a31b-18d83e6d9810",
    "jsonrpc":"2.0",
    "result": {
        "type": 0,
        "name": "Customer to Partner LTD",
        "id": "54aeab40b1a43dc0467b23e9",
        "address": "Str Example No 1",
        "phone": "0040740000001",
        "canBeManagedByAbove": true,
        "isSuspended": false
    }
}
```

2.1.4. suspendCompany

This method suspends an active company, disallowing this company's user accounts to log in to Control Center.

Parameters

Parameter	Type	Optional	Description
`companyId`	String	No	The ID of the company to be suspended.

Return value

This method does not return any value.

Example

Request :

```
{
     "params": {
        "companyId" : "5493dcd2b1a43df00b7b23c6"
     },
     "jsonrpc": "2.0",
     "method": "suspendCompany",
     "id": "8f6748b9-d201-4b63-b17f-4ecbebce24a9"
}
```

Response :

```
{
    "id":"8f6748b9-d201-4b63-b17f-4ecbebce24a9",
    "jsonrpc":"2.0",
    "result": null
}
```

2.1.5. activateCompany

This method activates a suspended company.

Parameters

Parameter	Type	Optional	Description
`companyId`	String	No	The ID of the company to be activated.

Return value

This method does not return any value.

Example

Request :

```
{
     "params": {
        "companyId" : "5493dcd2b1a43df00b7b23c6"
     },
     "jsonrpc": "2.0",
     "method": "activateCompany",
     "id": "c67860e2-36cc-43bd-bc0f-f1061c180b52"
}
```

Response :

```
{
    "id":"c67860e2-36cc-43bd-bc0f-f1061c180b52",
    "jsonrpc":"2.0",
    "result": null
}
```

2.2. Licensing

The Licensing API contains the following methods exposing the licensing related functionalities:

- `getLicenseInfo` : retrieves the license information for a company.
- `setMonthlySubscription` : enables the monthly subscription for a company.
- `setLicenseKey` : sets the license key for a company.

API endpoint: https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/licensing

2.2.1. getLicenseInfo

This method retrieves the license information for a company.

Parameters

Parameter	Type	Optional	Description
`companyId`	String	Yes	The ID of the company for which the license information is retrieved. The default value is the ID of the user's company.

Return value

This method returns an Object containing the license details:

- `subscriptionType` - the company's subscription type: 1 for TRIAL SUBSCRIPTION, 2 for LICENSED SUBSCRIPTION, 3 for INHERITED MONTHLY SUBSCRIPTION
- `expiryDate` - the license expiry date
- `usedSlots` - the number of used slots
- `totalSlots` - the number of total slots for LICENSED or INHERITED subscriptions.
- `licenseKey` - the license key for LICENSED subscription.

Example

Request :

```
{
     "params": {
        "companyId": "5493dcd2b1a43df00b7b23c6"
     },
     "jsonrpc": "2.0",
     "method": "getLicenseInfo",
     "id": "ad12cb61-52b3-4209-a87a-93a8530d91cb"
}
```

Response :

```
{
    "id":"c67860e2-36cc-43bd-bc0f-f1061c180b52",
    "jsonrpc":"2.0",
    "result": {
         "subscriptionType": 1,
         "expiryDate": "2015-01-18T10:02:30",
         "usedSlots": 0
     }
}
```

2.2.2. setMonthlySubscription

This method enables monthly subscription for a company.

Parameters

Parameter	Type	Optional	Description
`companyId`	String	No	The ID of the company that will inherit license seats from the parent company.
`reservedSlots`	Number	Yes	The number of slots to be reserved from the parent's company total amount of slots. If no value is passed, the maximum allowed number of slots will be selected.

Return value

This method does not return any value.

Example

Request :

```
{
     "params": {
        "companyId" : "5493dcd2b1a43df00b7b23c6",
        "reservedSlots" : 12
     },
     "jsonrpc": "2.0",
     "method": "setMonthlySubscription",
     "id": "d4d50719-3215-455a-a329-086fe77f6d72"
}
```

Response :

```
{
    "id":"2b5f52d8-5f6f-466c-b952-61b5c87c3182",
    "jsonrpc":"2.0",
    "result": null
}
```

2.2.3. setLicenseKey

This method sets the license key for a company.

Parameters

Parameter	Type	Optional	Description
`licenseKey`	String	No	The license key to be set.
`companyId`	String	Yes	The ID of the company whose license will be set. If no value is passed, the user's company will be selected.

Return value

This method does not return any value.

Example

Request :

```
    {
         "params": {
            "licenseKey" : "TNB3AAA",
            "companyId" : "5493dcd2b1a43df00b7b23c6"
         },
         "jsonrpc": "2.0",
         "method": "setLicenseKey",
         "id": "48daf1bc-4078-411c-bf44-4f293e68f501"
    }
```

Response :

```
    {
        "id":"48daf1bc-4078-411c-bf44-4f293e68f501",
        "jsonrpc":"2.0",
        "result": null
```

```
}
```

2.3. Network

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

- `getCompaniesList` : returns the list of companies under a parent company of from a company folder.
- `getCompanyFoldersList` : returns the list of company folders under a company or under a company folder.
- `createCompanyFolder` : creates a new folder under an existing folder or company.
- `moveCompanyOrCompanyFolder` : moves a company or company folder under a parent company or company folder.

API endpoint: https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/network

2.3.1. getCompaniesList

This method returns the list of companies under a parent company or from a company folder.

Parameters

Parameter	Type	Optional	Description
`parentId`	String	Yes	The parent company's ID or the company folder's ID. The default value is the ID of the parent company.

Return value

This method returns an Array containing the list of companies located under the parent. Each entry in the list has the following fields:

- `id` - the id of the company
- `name` - the name of the company

Example

Request :

```
{
     "params": {
         "parentId" : "54a28b41b1a43d89367b23fd"
```

```
        },
        "jsonrpc": "2.0",
        "method": "getCompaniesList",
        "id": "103d7b05-ec02-481b-9ed6-c07b97de2b7a"
    }
```

Response :

```
    {
        "id":"103d7b05-ec02-481b-9ed6-c07b97de2b7a",
        "jsonrpc":"2.0",
        "result": [
             {
                 "id" : "54a295eeb1a43d8b497b23c6",
                 "name" : "Customer Company"
             },
             {
                 "id" : "54a295d8b1a43d7c4a7b23c6",
                 "name" : "Partner Company"
             }
        ]
    }
```

2.3.2. getCompanyFoldersList

This method returns the list of company folders under a company or under a company folder.

Parameters

Parameter	Type	Optional	Description
`parentId`	String	Yes	The parent company's ID or the parent folder's ID. The default value is the ID of the parent company.

Return value

This method returns an Array containing the list of folders located under the parent. Each entry in the list has the following fields:

- `id` - the ID of the folder
- `name` - the name of the folder

Example

Request :

```
{
     "params": {
         "parentId" : "54a295d8b1a43d7c4a7b23c6"
     },
     "jsonrpc": "2.0",
     "method": "getCompanyFoldersList",
     "id": "8edf135b-f7cb-41f2-8b67-98054694f61e"
}
```

Response :

```
{
    "id":"8edf135b-f7cb-41f2-8b67-98054694f61e",
    "jsonrpc":"2.0",
    "result": [
            {
                "id" : "54a29726b1a43d8a497b23c6",
                "name" : "Company folder 1"
            },
            {
                "id" : "54a29746b1a43d0f4c7b23c6",
                "name" : "Company folder 2"
            }
        ]
}
```

2.3.3. createCompanyFolder

This method creates a new folder under an existing folder or company.

Parameters

Parameter	Type	Optional	Description
`folderName`	String	No	The name for the new folder.

Parameter	Type	Optional	Description
`parentId`	String	Yes	The parent folder's ID or the parent company's ID. The default value is the ID of the parent company.

Return value

This method returns a String: the id of the new created folder.

Example

Request :

```
{
     "params": {
         "parentId": "54a295d8b1a43d7c4a7b23c6",
         "folderName" : "Company folder 3"
     },
     "jsonrpc": "2.0",
     "method": "createCompanyFolder",
     "id": "f5911ea2-9f14-4046-96eb-5fa24cca98f0"
}
```

Response :

```
{
    "id":"f5911ea2-9f14-4046-96eb-5fa24cca98f0",
    "jsonrpc":"2.0",
    "result": "54a298c5b1a43d0c4c7b23c7"
}
```

2.3.4. moveCompanyOrCompanyFolder

This method moves a company or company folder under a parent company or company folder.

Parameters

Parameter	Type	Optional	Description
`id`	String	No	The ID of the company or company folder to be moved

Parameter	Type	Optional	Description
`newParentId`	String	No	The ID of the new parent company of company folder.

Return value

This method does not return any value.

Example

Request :

```
{
     "params": {
         "newParentId" : "54a29746b1a43d0f4c7b23c6",
         "id" : "54a298c5b1a43d0c4c7b23c7"
     },
     "jsonrpc": "2.0",
     "method": "moveCompanyOrCompanyFolder",
     "id": "7d2864e9-c67b-48a2-9ba3-0a11d47e83c8"
}
```

Response :

```
{
    "id":"7d2864e9-c67b-48a2-9ba3-0a11d47e83c8",
    "jsonrpc":"2.0",
    "result": null
}
```

2.4. Packages

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

- `getInstallationLinks` : returns the installation links available for managed companies.

API endpoint: https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/packages

2.4.1. getInstallationLinks

This method returns the installation links available for managed companies.

Parameters

Parameter	Type	Optional	Description
`companyId`	String	Yes	The ID of the managed company. If no value is passed, the installation links available for all managed companies will be returned.

Return value

This method returns an Array containing the list of installation links for each available package. Each entry in the list has the following fields:

- `packageName` - the name of the package
- `companyName` - the name of the company the package belongs to
- `companyId` - the id of the company the package belongs to
- `installLinkWindows` - the installation link for computers working with Windows
- `installLinkMac` - the installation link for computers working with MAC

Example

Request :

```
{
  "params": {
     "companyId": "54a1a1d3b1a43d2b347b23c1"
 },
  "jsonrpc": "2.0",
  "method": "getInstallationLinks",
  "id": "426db9bb-e92a-4824-a21b-bba6b62d0a18"
}
```

Response :

```
{
  "id": "426db9bb-e92a-4824-a21b-bba6b62d0a18",
  "jsonrpc": "2.0",
  "result": [{
      "packageName": "Pack1",
      "companyName": "TestC2",,
      "companyId": "54a1a1d3b1a43d2b347b23c1",
      "installLinkWindows":"https://gravityzone.bitdefender.com \
  /Packages/EPC/0/setupdownloader_[qwer=].exe",
      "installLinkMac":"https://gravityzone.bitdefender.com \
  /Packages/MAC/0/antivirus_for_mac_[qwer].pkg"
      }]
}
```

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 will create a company using C#.

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

String apiURL =
  "https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/";

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

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["type"] = 1;
parameters["name"] = "Test Company";
parameters["address"] = @"23rd Fictional Street,
  Hidden Town, Invisible Land";
parameters["phone"] = "+99000000000";
parameters["canBeManagedByAbove"] = true;

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

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

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

// Output:
// 54b646a0b1a43dbd197b23c6
```

3.2. curl Example

In the following example, we get information about all our managed companies using curl.

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

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":"54b646a0b1a43dbd197b23c6","name":"Test Company"}]}
```

3.3. Python Example

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

```
import base64
import pyjsonrpc
import requests
import simplejson

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

json = pyjsonrpc.create_request_json("getCompanyDetails",
  companyId="54b646a0b1a43dbd197b23c6")
result = requests.post(
  "https://gravityzone.bitdefender.com/api/v1.0/jsonrpc/companies",
  json,
  verify=False,
  headers = {
    "Content-Type": "application/json",
    "Authorization": authorizationHeader
  })

jsonResult = simplejson.loads(result.content)

print jsonResult

Output:

{u'jsonrpc': u'2.0',
u'id': u'28ffa0ca-a6bf-4153-b8e3-1ccb3443047d',
u'result': {u'name': u'Test Company', u'isSuspended': False,
u'phone': u'+9900000000',
u'address': u'23rd Fictional Street, Fantasy City, Dream land',
u'canBeManagedByAbove': True, u'type': 1,
u'id': u'54b646a0b1a43dbd197b23c6'}}
```

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://gravityzone.bitdefender.com/ \
    api/v1.0/jsonrpc/network",
  method: "POST",
  headers: {
    'Authorization':
      "Basic VWpsTVMrMG0xbDlJVVpqcGpXeUpHOGdibnYyTXRhNFQ6"
  },
  json: {
    "id": "123456789",
    "jsonrpc": "2.0",
    "method": "getCompaniesList",
    "params": []
  }
}, function(response, body) {
  console.log(body);
});

// Output:

// { id: '123456789', jsonrpc: '2.0',
//   result: [ { id: '54b646a0b1a43dbd197b23c6',
//   name: 'Test Company' } ] }
```