DEVTECH
DEVELOPING TECHNOLOGIES

Devtech cloud API

API documentation

DEVTECH LIMITED

27/03/2014

Contents

1 INTRODUCTION

The Devtech API provides an interface to allow you to access and perform actions on Devtech Cloud platform over external applications and scripts. The API is a HTTPS REST interface, REST (Representational State Transfer) and includes only POST based methods. Our API allows passing of parameters for POST methods only as JSON. API doesn’t have SSL server security present.

Devtech cloud API is accessible at http://challenge.devtechgroup.com/controllers/api.php

2 API DOCUMENTATION

For each API call there is only one parameter:

Parameter

name	value
data	**[JSON object]**

Value of this parameter is JSON object. In the description for each API call you will find how to properly structure JSON object.

2.1 Create Server

2.1.1 HTTP call

Description:

This function will create a server based on supplied parameters.

JSON object structure:

Attribute name	Type	Value
action	string	createServer
token	string	[User Token]
dataObject	CreateServer Model	[Object]

2.1.2 Response

Returns JSON object with following attributes

Attribute Name	Type	Description
success	bool	Depending on success of method returns **true** or **false**
message	string	Error message
data	Server Model	Server details

2.2 Create VNC connection to Server

2.2.1 HTTP Call

Description:
This method will create VNC (Virtual Network Computing) connection to server.

JSON object structure:

Attribute name	Type	Value
action	string	openVNCConnection
token	string	[User Token]
dataObject	OpenVNC Model	[Object]

2.2.2 Response

Returns JSON object with following attributes

Property name	Description
Success	True or False depending on the method success
Data	VNCResponse
Message	Error message/additional details

3 MODELS DOCUMENTATION

3.1 *Create Server Models*

3.1.1 CreateServer Model

Attribute Name	Type	Description
cpu (amount)	int	Amount of CPU
ram (in GB)	int	Amount of RAM memory in GB
hdd (in GB)	int	Size of HDD in GB
name	string	Server name

3.1.2 Server Model

Attribute Name	Type	Description
resourceType	string	Type of resource
resourceCreateDate	dateTime	The creation date of the resource
resourceName	string	The name of the resource
resourceState	string	The state of the resource
resourceUUID	string	The UUID of the resource (read-only)
customerName	string	The name of the customer that owns the resource
productOfferName	string	Product offer name
cpu	int	The number of virtual CPU cores assigned to the server
ram	int	The amount of memory assigned to
status	string	Server status
disks	Array[Disk Model]	A list of the disks attached to the server

3.1.3 Disk Model

Attribute Name	Type	Description
resourceType	string	Type of resource
resourceCreateDate	dateTime	The creation date of the resource
resourceName	string	The name of the resource
resourceState	string	The state of the resource
resourceUUID	string	The UUID of the resource (read-only)
customerName	string	The name of the customer that owns the resource
productOfferName	string	The name of the product offer
size	long	The capacity of the disk
status	string	The status of the disk
index	int	The index of the disk (zero-based) specifying its position within the server to which it is attached
serverName	string	The name of the server to which the disk is attached

3.2 Create VNC Models

3.2.1 OpenVNC Model

Attribute Name	Type	Description
serverID	string	**resourceUUID** for Server
username	string	VNC username
password	string	VNC password

3.2.2 VNCResponse

Attribute Name	Type	Description
url	string	url to Virtual Network Computing

4 EXAMPLE

Here is an example written in JQuery of POST request, which will write response object in console.log. You can use any programming language for writing your post requests.

```
<html>
<head>
<script src="http://code.jquery.com/jquery-1.11.0.min.js"></script>
<script>
        var request = {
                action:"getVersion",
                token : "70b02c03a1f95218f195e351e965b15c"
        };
        $.post("http://challenge.devtechgroup.com/controllers/api.php", {data: JSON.stringify(request)}, function(data){
                console.log(data);
        });
</script>
</head>
<body>
</body>
</html>
```