

Get Virtual Server Recipes

GET /virtual_machines/:virtual_machine_id/recipe_joins

Gets all recipes assigned to specified VS

API Essentials

API Key Generation

Autoscaling

- Add Autoscaling Rules
- Get List of Autoscaling Rules for VS
- Remove Autoscaling Rules

Backups

- Add/Edit Note
- Convert Backup to Template
- Create Backup
- Delete Backup
- Get List of All VS Backups
- Get List of Incremental Backups
- Get List of Normal Backups
- Restore Disk from Backup

Credit

- Get Credit

DNS

- Add DNS Record
- Add DNS Zone
- Delete DNS Record
- Delete DNS Zone
- Edit DNS Record
- Get DNS Zones
- Get List of DNS Zone Records
- Get List of Name Servers

Firewall Rules

- Add Firewall Rule
- Get Firewall Rules
- Update Firewall Rules

Flex Cloud VM Control

Getting Started with Superb Flex Cloud

IP Addresses

- Get IP Address Joins

Logs

- Get List of Log Items
- Get List of Transactions
- Get list of VS Transaction

Network Interfaces

Sections:

- Response Parameters
- JSON Request example
- JSON Response
- XML Request example

Response Parameters

compatible_with - recipe compatibility: *windows* or *unix*

created_at — the date when the recipe was created in the [YYYY][MM][DD]T[hh][mm][ss]Z format

description - recipe description

id - recipe ID

label - recipe label

script_type - script type for Windows-compatible recipes:

- *bat*
- *vbs*
- *powershell* (PowerShell v1.0)

updated_at — the date when the recipe was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format

use_on_hv_zones - true, if the recipe can be used on hypervisor zones, otherwise false

use_on_vms - true, if the recipe can be used on virtual servers, otherwise false

user_id — the ID of a recipe owner

recipe_steps - an array of recipe steps with the following details:

- *created_at* — the date when the step was created in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- *updated_at* — the date when the step was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- *fail_anything_else* - set true, if you have specified the recipe pass value, otherwise set false
You can only specify behavior for one scenario: for example, if the *fail_anything_else* = **false**, *pass_anything_else* must be set to **true**.
- *fail_values* - recipe fail value
- *failure_goto_step* - if the *on_failure* parameter = *go to step*, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.
- *id* - step ID
- *number* - step number
- *on_failure* - step behavior in case of failure:
 - *proceed* - proceed to the next step.
 - *fail* - terminate the recipe and mark it as failed.
 - *stop* - terminate the recipe and mark it as successful.
 - *go_to_step* - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.
- *on_success* - step behavior in case of success:
 - *proceed* - proceed to the next step.
 - *fail* - terminate the recipe and mark it as failed.
 - *stop* - terminate the recipe and mark it as successful.
 - *go_to_step* - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.
- *pass_anything_else* - set true, if you have specified the recipe fail value, otherwise set false
- *pass_values* - recipe pass value
- *recipe_id* - ID of a recipe the step belongs to

- Get VS Network Interfaces
- Rebuild VS Network

Recipes

- Add Recipe
- Add Recipe Step
- Assign Recipe to Virtual Server
- Delete Recipe
- Delete Recipe Step
- Edit Recipe
- Edit Recipe Step
- Get All Recipes
- Get Recipe Steps
- Get Virtual Server Recipes
- Remove recipe from Virtual Server
- Run Recipe on Multiple Virtual Servers
- Swap Recipe Step Number

- *result_source* - step result source:
 - *exit_code* - an exit status, e.g. 0 will be returned on success
 - *std_out* - standard output
 - *std_err* - standard error
 - *std_out_and_std_err* - standard output and standard error
- *success_goto_step* - if the *on_success* parameter = *go to step*, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

JSON Request example

```
curl -i -X POST -H "Accept: application/json"
-H "Content-type:application/json" -d
'{"recipe_join":{"recipe_id":"6",
"event_type":"vm_provisioning"}}' -u
user:userpass
<api_url>/virtual_machines/:virtual_machine_id/
recipe_joins.json
```

SSH Keys

- Add SSH Key
- Delete SSH Key
- Edit SSH Key
- Get SSH Keys
- Set SSH Keys on VS

Templates

- Get Templates

Test Route

Troubleshooting API Issues

Viewing Activity Logs

Virtual Server Operating Systems

Virtual Servers

- Add Virtual Server
- Billing Statistics
- Build Virtual Server
- Delete Virtual Server
- Edit Virtual Server
- Get CPU Usage Statistics
- Get List of Virtual Machines
 - Get specific VM Details
 - Search Virtual Servers by label
- Get statuses for All VMs
 - Get Specific VS Status
- Reboot Virtual Server
- Reset VS Password
- Shutdown Virtual Server
- Startup a Virtual Server
- Stop Virtual Server

JSON Response

```
{
  "vm_provisioning": [
    {
      "recipe": {
        "recipe_joins": {
          "compatible_with": "unix",
          "created_at":
"2015-08-29T00:34:33+00:00",
          "description": "Does
Stuff",
          "id": 8,
          "label": "Test Recipe",
          "script_type": "bat",
          "updated_at":
"2015-08-29T00:34:33+00:00",
          "user_id": 337,
          "recipe_steps": [
            {
              "recipe_step": {
                "created_at":
"2015-08-29T00:36:30+00:00",
                "fail_anything_else": true,
                "fail_values":
"",
                "failure_goto_step": null,
                "id": 2,
                "number": 1,
                "on_failure":
"proceed",
```

VS Disks

- Add New Disk
- Delete Disk
- Edit Disk
- Get VS Disks

```
        "on_success":
"proceed",
"pass_anything_else": false,
        "pass_values":
"Success",
        "recipe_id": 8,
"result_source": "exit_code",
        "script":
"#!/bin/bash\r\n# My first script\r\nnecho
\"Hello World!\"\r\n",
"success_goto_step": null,
        "updated_at":
"2015-08-29T00:36:30+00:00"
    }
    }
    ]
    },
    "join": {
        "recipe_joins": {
            "created_at":
"2015-08-31T18:32:02+00:00",
            "event_type":
"vm_provisioning",
            "id": 1,
            "recipe_id": 8,
            "target_join_id": 511,
            "target_join_type":
"VirtualMachine",
            "updated_at":
"2015-08-31T18:32:02+00:00"
        }
    }
}
```

```
}  
  ]  
}
```

XML Request example

```
curl -i -X POST -H "Accept: application/xml"  
-H "Content-type:application/xml" -d  
'<recipe_join><recipe_id>6</recipe_id><event_ty  
pe>vm_provisioning</event_type></recipe_join>'  
-u user:userpass  
<api_url>/virtual_machines/:virtual_machine_id/  
recipe_joins.xml
```