REST API

Show API Key | Reset API Key

Your VEID: 1079404
Your API KEY: please use button above to show

Whether you want to automatically create nightly snapshots, query status of your VPS, or develop a complete replacement for the KiwiVM panel, this page is for you.

All parameters can be passed either via GET or POST methods.

PHP examples

// Sample 1. Get information about server

```
$request = "https://api.64clouds.com/v1/getServiceInfo?veid=1079404&api_key=YOUR_API_KEY_HERE";
$serviceInfo = json_decode (file_get_contents ($request));
print_r ($serviceInfo);


/* ------------------------------ [ output ] ------------------------------
stdClass Object
(
    [hostname] => my.server.com
    [node_ip] => 10.20.30.40
    [node_alias] => Node32
    [node_location] => US, Florida
    [plan] => micro128
    [plan_monthly_data] => 322122547200
    [plan_disk] => 4294967296
    [plan_ram] => 155189248
    [plan_swap] => 37748736
    [os] => centos-6-x86_64
    [email] => customer@example.com
    [data_counter] => 569810827
    [data_next_reset] => 1430193600
    [ip_addresses] => Array
        (
            [0] => 11.22.33.44
            [1] => 11.22.33.45
        )

    [rdns_api_available] => 1
    [ptr] => stdClass Object
        (
            [11.22.33.44] => ns1.my.server.com
            [11.22.33.45] => ns2.my.server.com
        )

    [error] => 0
)
*/
```

// Sample 2. Create a snapshot

```
$request = "https://api.64clouds.com/v1/snapshot/create?description=Automatic_Snapshot&veid=1079404&api_key=YOUR_API_KEY_HERE";
$serviceInfo = json_decode (file_get_contents ($request));
print_r ($serviceInfo);

/* ------------------------------ [ output ] ------------------------------
stdClass Object
(
    [error] => 0
    [notificationEmail] => customer@example.com
)
*/
```

// Sample 3. Restart VPS

```
$request = "https://api.64clouds.com/v1/restart?veid=1079404&api_key=YOUR_API_KEY_HERE";
$serviceInfo = json_decode (file_get_contents ($request));
print_r ($serviceInfo);

/* ------------------------------- [ output ] -------------------------------
stdClass Object
(
    [error] => 0
)
*/
```

// Sample 4. Set PTR record

```
$request = "https://api.64clouds.com/v1/setPTR?ip=11.22.33.44&ptr=ns1.my.server.com&veid=1079404&api_key=YOUR_API_KEY_HERE";
$serviceInfo = json_decode (file_get_contents ($request));
print_r ($serviceInfo);

/* ------------------------------- [ output ] -------------------------------
stdClass Object
(
    [error] => 0
)
*/
```

// Sample 5. Restart VPS using wget

```
wget -qO- "https://api.64clouds.com/v1/restart?veid=1079404&api_key=YOUR_API_KEY_HERE"

/* ------------------------------- [ output ] -------------------------------
{"error":0}
*/
```

// Sample 6. Restart VPS using curl

```
//You may want to use curl instead as it allows passing all variables in a POST request

$requestData = array ("veid" => 1079404, "api_key" => "YOUR_API_KEY_HERE");
$request = "restart";
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, "https://api.64clouds.com/v1/$request");
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0); // curl running on Windows has issues with SSL -
                                             // see https://kb.ucla.edu/articles/how-do-i-use-curl-in-php-on-windows
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $requestData);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
$jsonData = curl_exec($ch);
if (curl_error($ch)) die("Connection Error: ".curl_errno($ch)." - ".curl_error($ch));
curl_close($ch);
print_r (json_decode ($jsonData));

/* ------------------------------- [ output ] -------------------------------
stdClass Object
(
    [error] => 0
)
*/
```

Available calls

Each API call requires that you supply a valid combination of VEID (VPS ID) and API key as shown in the example.

Each API call always returns *error* variable. If *error* is non-zero, check "message" variable for more details about the error.

Call	Parameters	Description and return values
start	none	Starts the VPS

stop	none	Stops the VPS
restart	none	Reboots the VPS
kill	none	Allows to forcibly stop a VPS that is stuck and cannot be stopped by normal means. Please use this feature with great care as any unsaved data will be lost.
getServiceInfo	none	*Returns* **vm_type:** Hypervizor type (ovz or kvm) **hostname:** Hostname of the VPS **node_ip:** IP address of the physical node **node_alias:** Internal nickname of the physical node **node_location:** Physical location (country, state) **location_ipv6_ready:** Whether IPv6 is supported at the current location **plan:** Name of plan **plan_disk:** Disk quota (bytes) **plan_ram:** RAM (bytes) **plan_swap:** SWAP (bytes) **os:** Operating system **email:** Primary e-mail address of the account **plan_monthly_data:** Allowed monthly data transfer (bytes). Needs to be multiplied by monthly_data_multiplier - see below. **data_counter:** Data transfer used in the current billing month. Needs to be multiplied by monthly_data_multiplier - see below. **monthly_data_multiplier:** Some locations offer more expensive bandwidth; this variable contains the bandwidth accounting coefficient. **data_next_reset:** Date and time of transfer counter reset (UNIX timestamp) **ip_addresses:** IPv4 and IPv6 addresses assigned to VPS (Array) **plan_max_ipv6s:** Maximum number of IPv6 addresses allowed by plan **rdns_api_available:** Whether or not rDNS records can be set via API **ptr:** rDNS records (Array of two-dimensional arrays: ip=>value) **suspended:** Whether VPS is suspended
getLiveServiceInfo	none	This function returns all data provided by getServiceInfo. In addition, it provides detailed status of the VPS. Please note that this call may take up to 15 seconds to complete. Depending on hypervisor this call will return the following information: *Returns* [OVZ hypervisor] **vz_status:** array containing OpenVZ beancounters, system load average, number of processes, open files, sockets, memory usage etc **vz_quota:** array containing OpenVZ disk size, inodes and usage info **is_cpu_throttled:** 0 = CPU is not throttled, 1 = CPU is throttled due to high usage. Throttling resets automatically every 2 hours. **ssh_port:** SSH port of the VPS *Returns* [KVM hypervisor] **ve_status:** Running or Stopped **ve_mac1:** MAC address of primary network interface **ve_used_disk_space_b:** Occupied (mapped) disk space in bytes **ve_disk_quota_gb:** Actual size of disk image in GB **is_cpu_throttled:** 0 = CPU is not throttled, 1 = CPU is throttled due to high usage. Throttling resets automatically every 2 hours. **ssh_port:** SSH port of the VPS (returned only if VPS is running) **live_hostname:** Result of "hostname" command executed inside VPS **load_average:** Raw load average string **mem_available_kb:** Amount of available RAM in KB **swap_total_kb:** Total amount of Swap in KB **swap_available_kb:** Amount of available Swap in KB

getAvailableOS	none	*Returns* **installed:** Currently installed Operating System **templates:** Array of available OS
reinstallOS	os	Reinstall the Operating System. OS must be specified via "os" variable. Use getAvailableOS call to get list of available systems.
resetRootPassword	none	Generates and sets a new root password. *Returns* **password:** New root password
getUsageGraphs	none	Obsolete, use getRawUsageStats instead
getRawUsageStats	none	Returns a two-dimensional array with the detailed usage statistics shown under Detailed Statistics in KiwiVM.
setHostname	newHostname	Sets new hostname.
setPTR	ip, ptr	Sets new PTR (rDNS) record for IP.
basicShell/cd	currentDir, newDir	Simulate change of directory inside of the VPS. Can be used to build a shell like Basic shell. *Returns* **pwd:** Result of the "pwd" command after the change.
basicShell/exec	command	Execute a shell command on the VPS (synchronously). *Returns* **error:** Exit status code of the executed command **message:** Console output of the executed command
shellScript/exec	script	Execute a shell script on the VPS (asynchronously). *Returns* **log:** Name of the output log file.
snapshot/create	description (optional)	Create snapshot *Returns* **notificationEmail:** E-mail address on file where notification will be sent once task is completed.
snapshot/list	none	Get list of snapshots. *Returns* **snapshots:** Array of snapshots (fileName, os, description, size, md5, sticky, purgesIn, downloadLink).
snapshot/delete	snapshot	Delete snapshot by fileName (can be retrieved with snapshot/list call).
snapshot/restore	snapshot	Restores snapshot by fileName (can be retrieved with snapshot/list call). This will overwrite all data on the VPS.
snapshot/toggleSticky	snapshot, sticky	Set or remove sticky attribute ("sticky" snapshots are never purged). Name of snapshot can be retrieved with snapshot/list call – look for fileName variable. Set sticky = 1 to set sticky attribute Set sticky = 0 to remove sticky attribute
snapshot/export	snapshot	Generates a token with which the snapshot can be transferred to another instance.
snapshot/import	sourceVeid, sourceToken	Imports a snapshot from another instance identified by VEID and Token. Both VEID and Token must be obtained from another instance beforehand with a snapshot/export

call.

ipv6/add	ip	Assigns a new IPv6 address. For initial IPv6 assignment an empty IP is required (call without parameters), and a new IP from the available pool is assigned automatically. All subsequent requested IPv6 addresses must be within the /64 subnet of the first IPv6 address. *Returns* **ip:** Newly assigned IPv6 address
ipv6/delete	ip	Releases specified IPv6 address.
migrate/getLocations	none	Return all possible migration locations. *Returns* **currentLocation:** ID of current location **locations:** IDs of locations available for migration into **descriptions:** Friendly descriptions of available locations **dataTransferMultipliers:** Some locations may offer more expensive bandwidth where monthly allowance will be lower. This array contains monthly data transfer allowance multipliers for each location.
migrate/start	location	Start VPS migration to new location. Takes new location ID as input. Note that this will result in all IPv4 addresses to be replaced with new ones, and all IPv6 addresses will be released. *Returns* **notificationEmail:** E-mail address on file where notification will be sent once task is completed. **newIps:** Array of new IP addresses assigned to the VPS.
cloneFromExternalServer	externalServerIP, externalServerSSHport, externalServerRootPassword	(OVZ only) Clone a remote server or VPS. See Migrate from another server for example on how this works.

getSuspensionDetails	none	Retrieve information related to service suspensions.
		
		Returns
		suspension_count: Number of times service was suspended in current calendar year
		suspensions: array of all outstanding issues along with supporing evidence of abuse. See example below.
		evidence: Full text of the complaint or more details about the issue
		
		Sample output when service is suspended:
		
		`[suspensions] => Array`
		`(`
		`[0] => stdClass Object`
		`(`
		`[record_id] => 11851 // Case ID, needed to unsuspend`
		`// the service via "unsuspend" API call`
		`[flag] => copyright // Type of abuse`
		`[is_soft] => 1 // 0 = must contact support to unsuspend`
		`// 1 = can unsuspend via API call`
		`[evidence_record_id] => 2207 // Detailed abuse report ID (see below)`
		`)`

```
    )

[evidence] => stdClass Object
    (
        [2207] => "Full text of abuse complaint here"
    )

[suspension_count] => 2
```

unsuspend	record_id	Clear abuse issue identified by record_id and unsuspend the VPS. Refer to getSuspensionDetails call for details.
getRateLimitStatus	none	When you perform too many API calls in a short amount of time, KiwiVM API may start dropping your requests for a few minutes. This call allows monitoring this matter. *Returns* **remaining_points_15min:** Number of "points" available to use in the current 15-minute interval **remaining_points_24h:** Number of "points" available to use in the current 24-hour interval