

Hashtopolis User API (V1)

s3in!c <s3inlc@hashes.org>

July 13, 2018

Introduction

The communication for the User API is always in JSON formatted values. When sending a request to the server, it should be a POST containing the JSON data. Every request has a *section* and a *request* field to state which action should be executed. Every response again then contains the requested section and request and gives a *status* to indicate if the query was successful or not. To increase the readability of this document, requests are always in blue, successful responses in green and error messages in red.

Errors

In case of an error with the query which the user sends to the server, the response will have following format with the corresponding action which was requested and the error message which should help in getting information about this error.

```
{
  "section":"task",
  "request":"create",
  "status":"ERROR",
  "message":"You are not allowed to create tasks!"
}
```

Sections

This part lists all sections available on the API. The value in the brackets denotes the according value to be sent on API queries.

- Access Control (*access*)
- Agents (*agent*)
- Server Config (*config*)
- Crackers (*cracker*)
- Files (*file*)
- Groups (*group*)

- Hashlists (*hashlist*)
- Preconfigured Tasks (*pretask*)
- Superhashlists (*superhaslist*)
- Supertasks (*supertask*)
- Tasks (*task*)
- Test (*test*)
- User Management (*user*)

Test (*test*)

This section is used to do testing queries, e.g. to test connectivity or availability of this API. The test section is the only one which allows to make requests without an access key.

connection

Used to test if the URL is a valid API endpoint.

```
{  
  "section": "test",  
  "request": "connection"  
}
```

```
{  
  "section": "test",  
  "request": "connection",  
  "response": "SUCCESS"  
}
```

access

Used to check if a given API key is still valid and can be used.

```
{  
  "section": "test",  
  "request": "access",  
  "accessKey": "mykey"  
}
```

```
{  
  "section": "test",  
  "request": "access",  
  "response": "OK"  
}
```

```
{  
  "section": "test",  
  "request": "access",  
  "response": "ERROR",  
  "message": "API key was not found!"  
}
```

Agents (*agent*)

Used to access all functions around agents. Please note that the user must have access to the groups where an agent is member of to retrieve it and to be able to apply changes.

setActive

Set an agent active/inactive.

```
{
  "section": "agent",
  "request": "setActive",
  "active": false,
  "agentId": 2,
  "accessKey": "mykey"
}

{
  "section": "agent",
  "request": "setActive",
  "response": "OK"
}
```

changeOwner

Either set an owner for an agent or remove the owner from it. The user can either be specified by providing the user ID or the username. If no owner should be specified, the user value must be *null*.

```
{
  "section": "agent",
  "request": "changeOwner",
  "user": 1,
  "agentId": 2,
  "accessKey": "mykey"
}

{
  "section": "agent",
  "request": "changeOwner",
  "user": "testuser",
  "agentId": 2,
  "accessKey": "mykey"
}
```

```
{  
  "section": "agent",  
  "request": "changeOwner",  
  "user": null,  
  "agentId": 2,  
  "accessKey": "mykey"  
}
```

```
{  
  "section": "agent",  
  "request": "changeOwner",  
  "response": "OK"  
}
```