

## Table of content

1. RELEASE NOTES .....	2
2. OVERVIEW .....	3
3. PRE-REQUEST .....	3
4. GENERAL SYNTAX.....	3
5. RESULT FORMAT .....	3
6. AUTHENTICATION .....	4
6.1. Authentication example .....	4
7. COMMANDS.....	5
7.1. CreateAccount .....	5
7.2. UpdateAccount .....	5
7.3. DeleteAccount .....	6
7.4. ActivateAccount.....	6
7.5. DisableAccount .....	6
7.6. AssignTokenToAccount.....	7
7.7. AccountIsActivated .....	7
7.8. GetAccountInfo.....	8
7.9. GetAccountStatus .....	9
7.10. GetAccountsList .....	9
7.11. GetLockedAccountsList.....	10
7.12. GetAccountProvisioningPdf .....	10
7.13. CheckOtp .....	10
7.14. SyncOtp.....	11
7.15. GetMemInfo .....	11
7.16. MasterSlaveFullResync .....	11

## History

### 2022-04-28 (5.8.7.0) Documentation enhanced

- 2018-08-21 (5.3.0.0) Documentation updated
- 2018-07-17 (5.2.0.3) Documentation enhanced
- 2017-05-16 (5.0.4.4) Additional command
- 2017-05-10 (5.0.4.0) Additional commands, enhanced results
- 2016-11-01 (5.0.2.6) Additional commands
- 2016-10-03 (5.0.2.3) Updated authentication mechanism
- 2016-05-03 (4.3.4.0) First public release

---

multiOTP® Enterprise is a strong two factors authentication appliance developed and maintained in Switzerland by SysCo systèmes de communication sa

Official website: <http://www.multiotp.com/>

---

## **1. Release notes**

### **2022-04-27 (5.8.7.0)**

ENH: Documentation enhanced

### **2018-08-21 (5.3.0.0)**

ENH: witout2FA algorithm added in the options of various commands

### **2018-07-17 (5.2.0.3)**

ENH: Documentation enhanced

### **2017-05-16 (5.0.4.4)**

ENH: New GetMemInfo command to retrieve appliance memory information

### **2017-05-10 (5.0.4.0)**

FIX: Numeric values are now all returned as numbers instead of string

ENH: New GetAccountProvisioningPdf command to get the provisioning PDF content of an account

ENH: New MasterSlaveFullResync command to force a full Master-Slave resynchronization

ENH: New GetAccountInfo command to retrieve all information about an account

ENH: New DisableAccount command replaces the DesactivateAccount

### **2016-11-01 (5.0.2.6)**

ENH: Documentation enhanced

### **2016-10-03 (5.0.2.3)**

ENH: Updated authentication mechanism, same as services like the Amazon AWS API REST authentication

### **2016-05-03 (4.3.4.0)**

First public release

## 2. Overview

This document explains how to use the Web API interface of **multiOTP® Enterprise**.

## 3. Pre-Request

Be sure that the Web API is used by a device for which Web API calls are allowed and enabled.

## 4. General syntax

The general syntax is the following:

`https://multiOTP_ip_address/?Command=command_string[&Arg1..n=value1..n]&Timestamp=ts&Signature=authentication_string`

## 5. Result format

Result is returned in JSON format. A general HTTP header code is also provided.

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    ...
  }
}
```

**Possible general HTTP header codes are:**

201: Created	400: Bad Request	500: Server error
202: Accepted	401: Unauthorized	
204: No content	404: Not Found	
	405: Method not allowed	

## 6. Authentication

The Web API authentication is now using the same signature mechanism as used for various services like the Amazon AWS API REST authentication, which is briefly described here:

- a) Add a Timestamp parameter with an URL encoded value of current GMT date/time in this format: **YYYY-MM-DDThh:mm:ssZ**
- b) Create the signature string:
  - 1) sort all parameters in the ascendant alphabetic order:
  - 2) Create the sorted request string with all parameters concatenated, separated by &
  - 3) Create the signature string (host, path (which is / in our case), request string: "GET\n`host_ip`\n`request_string`")
  - 4) Calculate the hash of the signature using the device secret :  
`urlencode(base64(hash_hmac_sha256(signature_string, device_secret)))`
- c) Add the Signature to the parameters and send the request to the multiOTP Enterprise device

### 6.1. Authentication example

multiOTP Enterprise virtual appliance: **192.168.1.88**

Secret defined for the Web API enabled device doing the request: **mysecret**

Command:

**Command=GetAccountStatus**

Parameter(s) with URL encoded value

**Account=demo\_account**

Add an URL encoded Timestamp parameter:

**Timestamp=2016-11-01T08%3A09%3A04Z**

Create the sorted request string for all parameters:

**Account=demo\_account&Command=GetAccountStatus&Timestamp=2016-11-01T08%3A09%3A04Z**

Create the signature string:

**GET\n192.168.1.88/\nAccount=demo\_account&Command=GetAccountStatus&Timestamp=2016-11-01T08%3A09%3A04Z**

Calculate the hash of the signature string: `urlencode(base64(hash_hmac_sha256("GET...", "mysecret")))`

**EqMMUOBpZwTosHOaFiaB82Ezv3UBE696W28jIWAQXkw%3D**

Send the full request which includes the signature to the multiOTP Enterprise virtual appliance:

**https://192.168.1.88/?Command=GetAccountStatus&Account=demo\_account&Timestamp=2016-11-01T08:09:04Z&Signature=EqMMUOBpZwTosHOaFiaB82Ezv3UBE696W28jIWAQXkw%3D**

Result returned (with a general HTTP header code of 202 Accepted):

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account",
    "activated": true,
    "delayed": false,
    "delayed_time": 0,
    "last_error": 1477184523,
    "locked": false,
    "user_last_login": 1477962123
  }
}
```

## 7. Commands

### 7.1. CreateAccount

Create a new account

#### Parameters:

Account: string, mandatory, name of the account.  
Activated: [0|1], optional, account activation. If not defined, the account is activated.  
Algorithm: [totp|hotp|motp|yubicootp|without2fa], optional, OTP algorithm of the account. If not defined, the default is totp.  
Description: string, optional, description of the account.  
Digits: integer, optional, number of digits used for the One Time Password. If not define, the default is 6.  
Email: string, optional, email address of the account.  
Group: string, optional, group(s) of the account. Several groups must be comma separated.  
LdapPassword: [-1|0|1], optional, using AD/LDAP password (1) instead of PIN (0), or take the system defined option (-1).  
NextEvent: integer, optional, next event for hotp or yubicootp algorithm. If not defined, the default value is 0.  
Pin: string, optional, PIN/password of the account. If not defined, a random 4 digits PIN is created.  
PrefixNeeded: [-1|0|1], optional, define if a prefix is needed (1) or not (0), or take the system defined option (-1).  
Seed: hexadecimal, optional, seed of the account. If not defined, a random seed is created.  
Sms: string, optional, mobile phone number of the account, in the international 00nnnnnnnnnnn format.  
Synchronized: [0|1], optional, account is AD/LDAP synchronized (1) or not. If not defined, the account is not synchronized.  
TimeInterval: integer, optional, time interval (in seconds) for totp algorithm. If not defined, the default value is 30.  
TokenAlgoSuite: [HMAC-SHA1], optional, token algorithm suite for totp/hotp, default is HMAC-SHA1.

#### Example:

```
https://192.168.1.88/?Command=CreateAccount&Account=demo_account&Algorithm=hotp&Description=demo%20account
&NextEvent=7&Pin=mypass&PrefixNeeded=1&Seed=3132333435363738393031323334353637383930
&Timestamp=2022-04-28T05:08:07Z&Signature=...
```

#### Result:

```
{
  "result_code": 11,
  "result_text": "INFO: User successfully created or updated",
  "result": {
    "account": "demo_account"
  }
}
```

### 7.2. UpdateAccount

Update existing account parameter(s).

#### Parameters:

Account: string, mandatory, name of the account.  
Activated: [0|1], optional, account activation.  
Algorithm: [totp|hotp|motp|yubicootp|without2FA], optional, OTP algorithm of the account.  
Description: string, optional, description of the account.  
Digits: integer, optional, number of digits used for the One Time Password.  
Email: string, optional, email address of the account.  
Group: string, optional, group(s) of the account. Several groups must be comma separated.  
LdapPassword: [-1|0|1], optional, using AD/LDAP password (1) instead of PIN (0), or take the system defined option (-1).  
NextEvent: integer, optional, next event for hotp or yubicootp algorithm.  
Pin: string, optional, PIN/password of the account.  
PrefixNeeded: [-1|0|1], optional, define if a prefix is needed (1) or not (0), or take the system defined option (-1).  
Seed: hexadecimal, optional, seed of the account.  
Sms: string, optional, mobile phone number of the account, in the international 00nnnnnnnnnnn format.  
Synchronized: [0|1], optional, account is AD/LDAP synchronized (1) or not (0).  
TimeInterval: integer, optional, time interval (in seconds) for totp algorithm.  
TokenAlgoSuite: [HMAC-SHA1], optional, token algorithm suite for totp/hotp.

#### Example:

```
https://192.168.1.88/?Command=UpdateAccount&Account=demo_account&Email=account@demo.com
&Timestamp=2022-04-28T05:08:07Z&Signature=...
```

#### Result:

```
{
  "result_code": 11,
  "result_text": "INFO: User successfully created or updated",
  "result": {
    "account": "demo_account"
  }
}
```

### 7.3. DeleteAccount

Delete an existing account.

**Parameter:**

Account: string, mandatory, name of the account.

**Example:**

[https://192.168.1.88/?Command=DeleteAccount&Account=demo\\_account&Timestamp=2022-04-28T05:08:07Z&Signature=...](https://192.168.1.88/?Command=DeleteAccount&Account=demo_account&Timestamp=2022-04-28T05:08:07Z&Signature=...)

**Result:**

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account"
  }
}
```

### 7.4. ActivateAccount

Activate an existing account.

**Parameter:**

Account: string, mandatory, name of the account.

**Example:**

[https://192.168.1.88/?Command=ActivateAccount&Account=demo\\_account&Timestamp=2022-04-28T05:08:07Z&Signature=...](https://192.168.1.88/?Command=ActivateAccount&Account=demo_account&Timestamp=2022-04-28T05:08:07Z&Signature=...)

**Result:**

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account"
  }
}
```

### 7.5. DisableAccount

Disable an existing account.

**Parameter:**

Account: string, mandatory, name of the account.

**Example:**

[https://192.168.1.88/?Command=DisableAccount&Account=demo\\_account&Timestamp=2017-05-10T05:00:04Z&Signature=...](https://192.168.1.88/?Command=DisableAccount&Account=demo_account&Timestamp=2017-05-10T05:00:04Z&Signature=...)

**Result:**

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account"
  }
}
```

## 7.6. AssignTokenToAccount

Assign a specific token to an account.

### Parameters:

Account: string, mandatory, name of the account.  
Token: string, mandatory, serial number of the token.

### Example:

`https://192.168.1.88/?Command=AssignTokenToAccount&Account=demo_account&Token=12345678&Timestamp=2022-04-28T05:08:07Z&Signature=...`

### Result:

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account",
    "token": "12345678"
  }
}
```

## 7.7. AccountIsActivated

Check if the account is activated.

### Parameter:

Account: string, mandatory, name of the account.

### Example:

`https://192.168.1.88/?Command=AccountIsActivated&Account=demo_account&Timestamp=2022-04-28T05:08:07Z&Signature=...`

### Result:

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account",
    "activated": true
  }
}
```

## 7.8. GetAccountInfo

Get account information.

### Parameter:

Account: string, mandatory, name of the account.

### Example:

[https://192.168.1.88/?Command=GetAccountInfo&Account=demo\\_account&Timestamp=2017-05-10T05:00:04Z&Signature=...](https://192.168.1.88/?Command=GetAccountInfo&Account=demo_account&Timestamp=2017-05-10T05:00:04Z&Signature=...)

### Result:

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account",
    "activated": true,
    "admin_with_otp": false,
    "algorithm": "hotp",
    "delayed": false,
    "delayed_finished": 0,
    "delayed_time": 0,
    "delta_time": 0,
    "description": "demo account",
    "dialin_ip_address": "",
    "dialin_ip_mask": "",
    "email": "",
    "error_counter": 0,
    "group": "",
    "key_id": "12345678",
    "language": "",
    "last_error": 0,
    "last_event": -1,
    "locked": false,
    "number_of_digits": 6,
    "request_ldap_pwd": false,
    "request_prefix_pin": true,
    "sms": "",
    "synchronized": false,
    "synchronized_channel": "",
    "synchronized_dn": "",
    "synchronized_server": "",
    "synchronized_time": 0,
    "time_interval": 0,
    "token_algo_suite": "HMAC-SHA1",
    "token_serial": "12345678",
    "user_last_login": 0,
    "welcome_mail_sent": false
  }
}
```



## 7.9. GetAccountStatus

Check the account status (activated, locked, delayed, delayed\_time, user\_last\_login, last\_error).

### Parameter:

Account: string, mandatory, name of the account.

### Example:

[https://192.168.1.88/?Command=GetAccountStatus&Account=demo\\_account&Timestamp=2022-04-28T05:08:07Z&Signature=...](https://192.168.1.88/?Command=GetAccountStatus&Account=demo_account&Timestamp=2022-04-28T05:08:07Z&Signature=...)

### Result:

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account",
    "activated": true,
    "delayed": false,
    "delayed_time": 0,
    "last_error": 1477184523,
    "locked": false,
    "user_last_login": 1477962123
  }
}
```

## 7.10. GetAccountsList

Get the list of all accounts.

### Parameter: n/a

### Example:

<https://192.168.1.88/?Command=GetAccountsList&Timestamp=2022-04-28T05:08:07Z&Signature=...>

### Result:

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": [
    {
      "account": "demo_account",
      "activated": true,
      "delayed": false,
      "description": "demo account",
      "group": "",
      "locked": false,
      "synchronized": false,
      "token": "12345678"
    },
    {
      "account": "second_user",
      "activated": true,
      "delayed": true,
      "delayed_finished": 1477988465,
      "delayed_time": 220,
      "description": "second user",
      "group": "",
      "locked": true,
      "synchronized": false,
      "token": ""
    }
  ]
}
```

## 7.11. GetLockedAccountsList

Get the list of all locked accounts.

**Parameter:** n/a

**Example:**

<https://192.168.1.88/?Command=GetLockedAccountsList&Timestamp=2022-04-28T05:08:07Z&Signature=...>

**Result:**

```
{
  "result_code": 19,
  "result_text": "INFO: Requested operation successfully done",
  "result": [
    {
      "account": "second_user",
      "activated": true,
      "delayed": true,
      "delayed_finished": 1477988465,
      "delayed_time": 220,
      "description": "second user",
      "group": "",
      "locked": true,
      "synchronized": false,
      "token": ""
    }
  ]
}
```

## 7.12. GetAccountProvisioningPdf

Get the provisioning PDF content of an account

**Parameter:**

Account: string, mandatory, name of the account.

**Example:**

[https://192.168.1.88/?Command=GetAccountProvisioningPdf&Account=demo\\_account&Timestamp=2017-05-10T05:00:04Z&Signature=...](https://192.168.1.88/?Command=GetAccountProvisioningPdf&Account=demo_account&Timestamp=2017-05-10T05:00:04Z&Signature=...)

**Result:**

The PDF content is returned, and the returned http header code is 202 in case of success, and 404 otherwise.

## 7.13. CheckOtp

Check the OTP of an account.

**Parameters:**

Account: string, mandatory, name of the account.

Otp: string, mandatory, OTP for this account.

Otp2: string, optional, second OTP for this account (used to unlock and check at the same time).

Prefix: string, optional, PIN or AD/LDAP password prefix, if needed for this account.

**Example:**

[https://192.168.1.88/?Command=CheckOtp&Account=demo\\_account&Otp=287922&Prefix=mypass&Timestamp=2022-04-28T05:08:07Z&Signature=...](https://192.168.1.88/?Command=CheckOtp&Account=demo_account&Otp=287922&Prefix=mypass&Timestamp=2022-04-28T05:08:07Z&Signature=...)

**Result:**

```
{
  "result_code": 0,
  "result_text": "OK: Token accepted",
  "result": {
    "account": "demo_account"
  }
}
```

## 7.14. SyncOtp

Synchronize the OTP of an account.

### Parameters:

Account: string, mandatory, name of the account.  
Otp: string, mandatory, OTP for this account.  
Otp2: string, mandatory, second OTP for this account.  
Prefix: string, optional, PIN or AD/LDAP password prefix, if needed for this account.

### Example:

`https://192.168.1.88/?Command=CheckOtp&Account=demo_account&Otp=399871&Otp2=520489&Prefix=mypass&Timestamp=2022-04-28T05:08:07Z&Signature=...`

### Result:

```
{
  "result_code": 14,
  "result_text": "INFO: Token has been resynchronized successfully",
  "result": {
    "account": "demo_account"
  }
}
```

## 7.15. GetMemInfo

Get appliance memory information

Parameter: n/a

### Example:

`https://192.168.1.88/?Command=GetMemInfo&Timestamp=2017-05-16T05:00:04Z&Signature=...`

### Result:

```
{
  "result_code": 14,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "MemTotal": "4144692 kB",
    "MemFree": "3151504 kB",
    "MemAvailable": "3497112 kB",
    ...
  }
}
```

## 7.16. MasterSlaveFullResync

Force a full Master-Slave resynchronization.

Parameter: n/a

### Example:

`https://192.168.1.88/?Command=MasterSlaveFullResync&Timestamp=2017-05-10T05:00:04Z&Signature=...`

### Result:

```
{
  "result_code": 14,
  "result_text": "INFO: Requested operation successfully done",
  "result": {
    "account": "demo_account"
  }
}
```