

# Guide to cPanel API 1

## Guide to cPanel API 1

Return Data

cPanel Tags  
(deprecated)

Bandwidth:  
displayheader

Bandwidth::displaybw

Bandwidth::displayday

Bandwidth::displaymainbwheder

### BoxTrapper

BoxTrapper::changestatus

BoxTrapper::cleancfgfilelist

BoxTrapper::editmsg

BoxTrapper::fetchcfgfile

BoxTrapper::getboxconfdir

BoxTrapper::listmsgsgs

BoxTrapper::logcontrols

BoxTrapper::messagereaction

BoxTrapper::multimessagereaction

BoxTrapper::resetmsg

BoxTrapper::savecfgfile

BoxTrapper::saveconf

BoxTrapper::showautowhitelist

BoxTrapper::showemails

BoxTrapper::showfromname

BoxTrapper::showlog

## Introduction

cPanel API 1 accesses information from the cPanel interface.

### Warning:

cPanel API 1 is **deprecated**. We **strongly** recommend that you use cPanel API 2 or UAPI instead.

### Important:

In cPanel & WHM version 76, we introduced server profiles and roles. When you enable a non-*Standard Node* server profile, the system **disables** API calls associated with that profile's disabled roles.

### Note:

You **cannot** call this API's functions via another API's call methods. For more information, read the [API method distinctions](#) section below.

## Basic usage

[LiveAPI](#) [PHP Class](#) [LiveAPI Perl Module](#) [Template Toolkit](#)  
cPanel Tag System (deprecated) [Command Line](#)

```
$cpanel = new CPANEL(); // Connect to  
cPanel - only do this once.  
$your_variable = $cpanel->api('Module',  
'function', array('parameter',  
'parameter', 'parameter')); // Call the  
function.
```

### Note:

For more information, read our [Guide to the LiveAPI System](#).

```
my $cpliveapi = Cpanel::LiveAPI->new();  
# Connect to cPanel - only do this once.  
my $your_variable =  
$cpliveapi->api('Module', 'function',  
['parameter', 'parameter', 'parameter']  
); # Call the function.
```

### Note:

For more information, read our [Guide to the LiveAPI System](#).

## cPanel API 1 Basics

## About cPanel API 1

cPanel API 1 accesses information about cPanel accounts.

**Warning:** cPanel API 1 is **deprecated**. We **strongly** recommend that you use cPanel API 2 or UAPI instead.

- BoxTrapper:: show message
- BoxTrapper:: show queue
- BoxTrapper:: show queuesearch
- BoxTrapper:: show queuetime
- BoxTrapper:: status
- BoxTrapper:: statusbutton
- ▼ Branding
  - Branding:: file
  - Branding:: image
  - Branding:: killing
  - Branding:: setbrandingpkg
  - Branding:: setmyacctspkg
  - Branding:: setupdirs
  - Branding:: tempsetbrandingpkg
- ▼ Cgi
  - Cgi:: backuplink
  - Cgi:: displayurchin
  - Cgi:: phpmyadminlink
- ▼ Chkserverd
  - Chkserverd:: getexport
  - Chkserverd:: getexport\_ssl
- ▼ ClamScanner
  - ClamScanner:: bars
- ▼ cPAddons
  - cPAddons:: mainpg
- ▼ Cron
  - Cron:: crontab\_perms
- ▼ CustInfo
  - CustInfo:: getemail
- ▼ Denylp
  - Denylp:: adddenyip
  - Denylp:: deldenyip
  - Denylp:: listdenyips

**Note:**

Currently, individual function documentation does **not** include Template Toolkit examples.

```

[%-
USE Apil;
SET myvariable = execute(
    'Module', 'function',
    {
        'parameter' => 'value',
        'parameter' => 'value',
        'parameter' => 'value',
    }
);
%]
```

**Note:**

For more information, read our [Guide to Template Toolkit documentation](#).

**Warning:**

In cPanel & WHM version 11.30 and later, cPanel tags are **deprecated**. We provide this example to help developers move from the old cPanel tag system to our [LiveAPI](#) system. We **strongly** recommend that you **only** use the [LiveAPI](#) system to call the cPanel APIs.

```

<cpanel Module="Module::function(
$FORM('parameter', 'parameter', 'parameter') )">
```

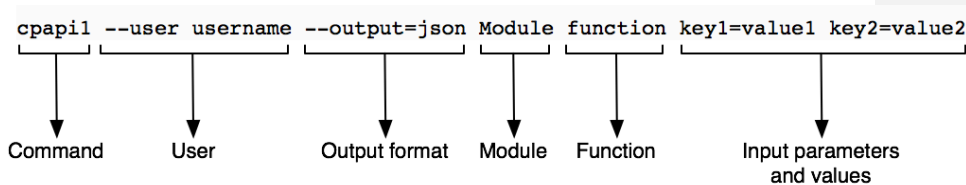
```

cpapil --user=username --output=type
Module function parameter=value
parameter=value
```

**Note:**

cPanel & WHM introduced the command line feature in version 56.

cPanel API 1 calls through the command line consist of the following basic parts:



**Notes:**

- Th
- Fe
- int
- Ur
- mu
- us
- 20
- or
- 20
- 20
- or
- 20
- ).

**Related Documentation**

- Use WHM API to Call cPanel API and UAPI — You can call cPanel API and UAPI functions through the WHM API.



Email:: getarsstart  
 Email:: getarsstop  
 Email:: getarssubject  
 Email:: getmailserver  
 Email:: getmailserveruser  
 Email:: getpopquota  
 Email:: has\_spam\_as\_acl  
 Email:: has\_spam\_autodelete  
 Email:: listmaildomains  
 Email:: listmaildomainsopt  
 Email:: listmaildomainsoptndef  
 Email:: mainacctdiskused  
 Email:: passwdlist  
 Email:: passwdpop  
 Email:: setmxaccept  
 Email:: spamboxstatus  
 Email:: spamstatus  
 Email:: tracefilter

▼ **Fileman**

Fileman:: changeperm  
 Fileman:: delfile  
 Fileman:: dofileop  
 Fileman:: emptytrash  
 Fileman:: extractfile  
 Fileman:: fileimg  
 Fileman:: fmmkdir  
 Fileman:: fmmkfile  
 Fileman:: fmpushfile  
 Fileman:: fmrename  
 Fileman:: fmsavefile  
 Fileman:: fmsaveht

<i>Output Type</i>	<p>The API output type that you wish to receive.</p> <ul style="list-style-type: none"> <li>• Use <code>--output=json</code> to return JSON-formatted output.</li> <li>• Use <code>--output=jsonpretty</code> to return indented JSON-formatted output.</li> <li>• Use <code>--output=xml</code> to return XML-formatted output.</li> </ul> <div data-bbox="860 640 1182 886" style="border: 1px solid red; padding: 5px; text-align: center;"> <p><b>Important:</b>        cPanel &amp; WHM version 66 <b>depre</b>  <b>cated</b> XML        output.</p> </div> <ul style="list-style-type: none"> <li>• Use <code>--output=yaml</code> to return YAML-formatted output.</li> </ul> <div data-bbox="820 1060 1182 1270" style="border: 1px solid orange; padding: 5px; text-align: center;"> <p><b>Note:</b>        This parameter        defaults to <code>--output=yaml</code>.</p> </div>
<i>Module</i>	The cPanel API 1 module name.
<i>Function</i>	The cPanel API 1 function.
<i>User</i>	The cPanel account-level username.
<i>Input parameters and values</i>	The function's input parameters and their values.

- [Guide to cPanel API 2 — cPanel API 2 access cPanel account information and modifies settings.](#)

mfile

Fileman:: fullbacku  
p

Fileman:: getfile

Fileman:: htmledito  
r

Fileman:: killdir

Fileman:: listfiles

Fileman:: listfullba  
ckups

Fileman:: listtempl  
ates

Fileman:: restoreaf

Fileman:: restored  
b

Fileman:: restorefil  
e

Fileman:: restorefil  
es

Fileman:: savehtml  
file

Fileman:: showper  
m

Fileman:: uploadfil  
es

Fileman:: viewfile

▼ Frontpage

FrontPage:: addfp

FrontPage:: delfp

FrontPage:: fpstat  
us

▼ Ftp

Ftp:: addftp

Ftp:: delftp

Ftp:: ftpquota

Ftp:: ftpservernam  
e

Ftp:: ftpstatslogin

Ftp:: get\_anonftp

Ftp:: get\_anonftp  
pin

Ftp:: get\_welcome  
msg

Ftp:: getftpquota

Ftp:: kill\_ftp\_sessi  
on

Ftp:: listftpstatsurl

Ftp:: passwdftp

- You **must** [URI-encode](#) va  
lues.
- Separate multiple param  
eter=value pairs with a  
space character.
- Special characters within  
a key's value may cause  
an error. You **must** either  
escape any special  
characters within values  
or surround the value with  
appropriate quotes. For  
more information, read  
Wikipedia's [Escape  
Characters](#) article.  
For example, a bash shell  
command with a  
JSON-encoded value  
may appear similar to one  
of the following:

- ```
cpapil --user  
username Module  
function  
key=[\"sslinstal  
l\", \"videotut\"  
]
```
- ```
cpapil --user  
username Module  
function key='{  
videotut\", \"ssl  
install\"}'
```

**Note:**

The term "Boolean" in  
our documentation  
refers to parameters  
that accept values of  
1 or 0. cPanel &  
WHM's APIs do **not** s  
upport the literal  
values of `true` and `f`  
alse.

- Ftp:: set\_anonftp
- Ftp:: set\_anonftpin
- Ftp:: set\_welcome  
msg
- ▼ Gpg
  - Gpg:: deletekey
  - Gpg:: exportkey
  - Gpg:: exportsecret  
key
  - Gpg:: genkey
  - Gpg:: importkeys

--	--

**Note:**  
For more information about this feature, run the following command:

```
cpapil --help
```

These examples use the following variables:

Variable	Description	Example
Module	The module for the function that you wish to use.	Email
function	The cPanel API 1 function that you wish to use.	addpop

- ▼ Htaccess
  - Htaccess:: checkpr  
otected
  - Htaccess:: del\_use  
r
  - Htaccess:: getinde  
x
  - Htaccess:: htdirls
  - Htaccess:: number  
\_of\_users
  - Htaccess:: phpsele  
ctable
  - Htaccess:: resnam  
e
  - Htaccess:: set\_pas  
s
  - Htaccess:: set\_pro  
tect
  - Htaccess:: setinde  
x
  - Htaccess:: setphpr  
eference
  - Htaccess:: showus  
ers
- ▼ ImageManager
  - ImageManager:: c  
onvert
  - ImageManager:: di  
mensions
  - ImageManager:: h  
dimension
  - ImageManager:: s  
cale
  - ImageManager:: th  
umbnail
  - ImageManager:: w  
dimension
- ▼ LastLogin
  - LastLogin:: lastlogi  
n

- LeechProtect:: disable
  - LeechProtect:: enable
  - LeechProtect:: setup
  - LeechProtect:: showpasswdfile
  - LeechProtect:: status
- LogManager:: listdownloads
  - LogManager:: save settings
  - LogManager:: show settings
- MagicRevision:: uri
- Mime:: add\_handler
  - Mime:: add\_hotlink
  - Mime:: add\_mime
  - Mime:: add\_redirect
  - Mime:: del\_handler
  - Mime:: del\_hotlink
  - Mime:: del\_mime
  - Mime:: del\_redirect
  - Mime:: linkallownull
  - Mime:: linkenable
  - Mime:: linkexts
  - Mime:: linkrurl
  - Mime:: linkurls
- Mysql:: adddb
  - Mysql:: addhost
  - Mysql:: adduser
  - Mysql:: adduserdb
  - Mysql:: checkdb
  - Mysql:: deldb
  - Mysql:: delhost
  - Mysql:: deluser
  - Mysql:: deluserdb

parameter	Input parameters that you pass to the function.	example.com
	<p><b>Important:</b></p> <ul style="list-style-type: none"> <li>cPanel API 1 functions use ordered parameters. You <b>must</b> specify parameters in the order in which they appear in this documentation.</li> <li>cPanel API 1 parameters are <b>case-sensitive</b>.</li> </ul>	
	<p><b>Note:</b></p> <p>The term "Boolean" in our documentation refers to parameters that accept values of 1 or 0. cPanel &amp; WHM's APIs do <b>not</b> support the literal values of true and false.</p>	

**Warning:**  
Do **not** attempt to use cPanel or WHM interface URLs to perform actions in custom code. You **must** call the appropriate API functions in order to perform the actions of cPanel & WHM's interfaces.

For example, do **not** pass values to .html pages, as in the following example:

```
http://example.com:2082/frontend/x3/mail/doaddpop.html/email=name&domain=user.com
```

While this **unsupported** method sometimes worked in previous versions of cPanel & WHM, we **strongly** discourage its use and do **not** guarantee that it will work in the future. Instead, the correct method to perform this action is to call the appropriate API function.

- Mysql:: gethost
- Mysql:: initcache
- Mysql:: listdbs
- Mysql:: listdbsopt
- Mysql:: listusers
- Mysql:: listusersopt
- Mysql:: number\_of\_db
- Mysql:: number\_of\_users
- Mysql:: repairedb
- Mysql:: routines
- Mysql:: updateprivs
- Net
  - Net:: dnslookup
  - Net:: dnszone
  - Net:: sethastracero
- News
  - News:: displaynews
- NVData
  - NVData:: set
- OptimizeWS
  - OptimizeWS:: load\_optimizsettings
  - OptimizeWS:: load\_optimizsettings (bytes)
  - OptimizeWS:: optimizews
- Park
  - Park:: disabledirect
  - Park:: getredirecturl
  - Park:: park
  - Park:: setredirecturl
  - Park:: unpark
- Postgres
  - Postgres:: adddb
  - Postgres:: adduser
  - Postgres:: adduserdb
  - Postgres:: deldb
  - Postgres:: deluser

## Call cPanel API 1 from the WHM API

**Warning:**  
Unless you call this API via the WHM API, you **cannot** call this API via WHM's ports (2086 or 2087).

You can use the WHM API to call cPanel API functions. For more information, read our [Use WHM API to Call cPanel API & UAPI](#) documentation.

**Important:**  
Because some cPanel API 1 functions return HTML output, this method will cause parsing errors in most XML parsers. We **strongly** recommend that you use cPanel API 2 instead.

## Custom Event Handlers and Standardized Hooks

You can use the [Standardized Hook System](#) to customize the events before and after a cPanel API call.

## API method distinctions

cPanel, L.L.C. produces four current and two deprecated APIs, and they all include separate sets of functions. Make **certain** that the function and module that you call exist in the API version that your code uses.

**Important:**  
API calls **must** use the correct port:

- 2082 — Unsecure calls to cPanel's APIs.
- 2083 — Secure calls to cPanel's APIs.
- 2095 — Unsecure calls to cPanel's APIs via a Webmail session.
- 2096 — Secure calls to cPanel's APIs via a Webmail session.
- 2086 — Unsecure calls to WHM's APIs, or to cPanel's APIs via the WHM API.
- 2087 — Secure calls to WHM's APIs, or to cPanel's APIs via the WHM API.

Otherwise-correct calls will return `Permission denied` or `Function not found` errors if they use an incorrect port number.

Select a tab to view examples of calls to the desired API:  
[WHM API 1](#) [UAPI](#) [cPanel API 2](#) [WHM API 0 \(deprecated\)](#)  
[cPanel API 1 \(deprecated\)](#)

### Browser-based call...

```
https://hostname.example.com:2087/cpess#####/json-api/accountssummary?api.version=1&user=username
```

**Note:**  
Browser-based calls to WHM API 1 **must** include the WHM API version (`api.version=1`). If you omit the version, the system calls WHM API



Postgres:: deluser  
db

Postgres:: initcach  
e

Postgres:: listdbso  
pt

Postgres:: listusers  
opt

Postgres:: number  
\_of\_dbs

Postgres:: number  
\_of\_users

#### ProxyUtils

ProxyUtils:: proxya  
ddress

#### Resellers

Resellers:: accoun  
tlistopt

Resellers:: themeli  
stopt

#### Serverinfo

Serverinfo:: servic  
estatus

#### SetLang

SetLang:: listlang  
opt

SetLang:: setlang

#### SpamAssassin

SpamAssassin:: co  
nfig

SpamAssassin:: sa  
veconfig

#### SSH

SSH:: getport

#### SSL

SSL:: delete

SSL:: deletectr

SSL:: deletecsr

SSL:: deletekey

SSL:: genctr

SSL:: gencsr

SSL:: genkey

SSL:: getcabundle

SSL:: getcname

SSL:: install

SSL:: installedhost

SSL:: listkeysopt

SSL:: showcrt

0, which may not contain the desired function, or may contain an older version of that function.

#### Command-line call...

```
whmapi1 accountssummary user=username
```

For more information, read our [Guide to Testing Custom Code - WHM API Calls](#) and [Guide to WHM API 1](#) documentation.

#### cPanel or Webmail session URL call...

```
https://hostname.example.com:2083/cps  
ess#####/execute/Module/function  
?parameter=value&parameter=value&para  
meter=value
```

#### Command-line call...

```
uapi --user=username --output=type  
Module function parameter=value  
parameter=value
```

#### LiveAPI PHP Class call...

```
$cpanel = new CPANEL(); // Connect to  
cPanel - only do this once.  
  
// Call a UAPI function.  
$function_result = $cpanel->uapi(  
    'Module', 'function',  
    array(  
        'parameter' => 'value',  
        'parameter' => 'value',  
        'parameter' => 'value',  
    )  
);
```

#### LiveAPI Perl Module call...

- SSL:: showcsr
- SSL:: showkey
- SSL:: sslsharedstatus
- SSL:: uploadcrt
- SSL:: uploadkey
- StatManager:: doForm
  - StatManager:: updateUserConfig
- Stats:: analoglist
  - Stats:: awstats
  - Stats:: bwbar
  - Stats:: errlog
  - Stats:: lastvisitors
  - Stats:: showbandwidth
  - Stats:: webalizer
  - Stats:: webalizerftp
- SubDomain:: addsubdomain
  - SubDomain:: cplistsubdomains
  - SubDomain:: disablesubrd
  - SubDomain:: listsubdomainsop
  - SubDomain:: setsuburl
- UI:: confirm
  - UI:: redirect
  - UI:: showresult

```

my $cpliveapi =
Cpanel::LiveAPI->new(); # Connect to
cPanel - only do this once.

# Call a UAPI function.
my $function_result =
$cpliveapi->uapi(
    'Module', 'function',
    {
        'parameter' => 'value',
        'parameter' => 'value',
        'parameter' => 'value',
    }
);

```

#### Template Toolkit call...

```

<!-- Call a UAPI function. -->
[%
execute(
    'Module', 'function',
    {
        'parameter' => 'value',
        'parameter' => 'value',
        'parameter' => 'value',
    }
);
%]

```

#### Note:

For more information, read our [Guide to Testing Custom Code - cPanel API and UAPI Calls](#) and [Guide to UAPI](#) documentation.

#### Call through the WHM API...

```

https://hostname.example.com:2087/cps
ess#####/json-api/cpanel?cpanel_
jsonapi_user=user&cpanel_jsonapi_apiv
ersion=2&cpanel_jsonapi_module=Module
&cpanel_jsonapi_func=function&paramet
er="value"

```

#### Command-line call...

```

cpapi2 --user=username --output=type
Module function parameter=value
parameter=value

```

#### ▼ Template Toolkit call...

```
<!-- Call a cPanel API 2 function.
-->
[%-
USE Api2;
SET myvariable = execute(
    'Module', 'function',
    {
        'parameter'    => 'value',
        'parameter'    => 'value',
        'parameter'    => 'value',
    }
);
%]
```

#### ▼ LiveAPI PHP Class call...

```
$cpanel = new CPANEL(); // Connect to
cPanel - only do this once.

// Call the function.
$my_variable = $cpanel->api2(
    'Module', 'function',
    array(
        'parameter'    => 'value',
        'parameter'    => 'value',
        'parameter'    => 'value',
    )
);
```

#### ▼ LiveAPI Perl Module call...

```
my $cpliveapi =
Cpanel::LiveAPI->new(); # Connect to
cPanel - only do this once.

# Call the function.
my $my_variable = $cpliveapi->api2(
    'Module', 'function',
    {
        'parameter'    => 'value',
        'parameter'    => 'value',
        'parameter'    => 'value',
    }
);
```

**Note:**

For more information, read our [Guide to Testing Custom Code - cPanel API](#)

and UAPI Calls and Guide to cPanel API 2 documentation.

**Warning:**

WHM API 0 is **deprecated**. We **strongly** recommend that you **only** use the equivalent WHM API 1 functions instead.

▼ Browser-based call...

```
https://hostname.example.com:2087/cps  
ess#####/json-api/accountssummary  
?user=username
```

▼ Command-line call...

```
whmapi0 accountssummary user=username
```

**Note:**

For more information, read our [Guide to Testing Custom Code - WHM API Calls](#) and [Guide to WHM API 0 documentation](#).

**Warning:**

cPanel API 1 is **deprecated**. We **strongly** recommend that you use cPanel API 2 or UAPI instead.

▼ Command-line call...

```
cpapi1 --user=username --output=type  
Module function parameter=value  
parameter=value
```

▼ LiveAPI PHP Class call...

```
$cpanel = new CPANEL(); // Connect to  
cPanel - only do this once.  
$your_variable =  
$cpanel->apil('Module', 'function',  
array('parameter', 'parameter',  
'parameter') ); // Call the function.
```

▼ LiveAPI Perl Module call...

```
my $cpliveapi =  
Cpanel::LiveAPI->new(); # Connect to  
cPanel - only do this once.  
my $your_variable =  
$cpliveapi->apil('Module',  
'function', ['parameter',  
'parameter', 'parameter'] ); # Call  
the function.
```

#### ▼ Template Toolkit call...

```
<!-- Call a cPanel API 1 function.  
-->  
[%-  
USE Apil;  
SET myvariable = execute(  
    'Module', 'function',  
    {  
        'parameter' => 'value',  
        'parameter' => 'value',  
        'parameter' => 'value',  
    }  
);  
%]
```

**Note:**

For more information, read our [Guide to Testing Custom Code - cPanel API and UAPI Calls](#) and [Guide to cPanel API 1 documentation](#).