

Guide to UAPI

Guide to UAPI

Return Data
Filter Output
Sort Output
Paginate Output
Output Columns
Calls from the WHM API
Custom UAPI Modules

Backup
Backup::
fullbacku

p_to_ftp

Backup::
fullbacku
p_to_homedir

Backup::
fullbacku
p_to_scp_with_key

Backup::
fullbacku
p_to_scp_with_password

Backup::
list_backups

Bandwidth

Bandwidth::
get_enabled_protocols

Bandwidth::
get_retention_periods

Bandwidth::
query

Batch

Batch::
strict

Brand

Brand::
read

Branding

Branding::
file

Branding::
get_applications

Branding::
get_application_information

Branding::
get_available_applications

Branding::
get_information_for_applications

Introduction

UAPI accesses the cPanel interface's features. Use this API to access and modify cPanel account data and settings.

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.

Notes:

- If you encounter problems with a local access method (LiveAPI PHP, LiveAPI Perl, or Template Toolkit), attempt a [browser-based call](#).
- 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

cPanel or Webmail Session URLLiveAPI PHP ClassLiveAPI Perl Module
cPanel Template ToolkitCommand Line

```
/execute/Module/function?parameter=value  
&parameter=value&parameter=value
```

```
$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',  
    )  
);
```

Note:

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

UAPI Basics

About UAPI

UAPI accesses and modifies data for cPanel accounts.

Notes:

- Th
Fe

int
Ur
mi
us
20
or
20
)
'
20
or
20
)

- Branding:: image
- Branding:: include
- Branding:: spritelist
- CacheBuster
 - CacheBuster:: read
 - CacheBuster:: update
- Chkserverd
 - Chkserverd:: get_exim_ports
 - Chkserverd:: get_exim_ports_ssl
- Chrome
 - Chrome:: get_dom
- Contactus
 - Contactus:: is_enabled
- cPAddons
 - cPAddons:: get_available_addons
 - cPAddons:: get_instance_settings
 - cPAddons:: list_addon_instances
- cPGreyList
 - cPGreyList:: disable_all_domains
 - cPGreyList:: disable_domains
 - cPGreyList:: enable_all_domains
 - cPGreyList:: enable_domains
 - cPGreyList:: has_greylisting_enabled
 - cPGreyList:: list_domains
- DAV
 - DAV:: disable_shared_global_addressbook
 - DAV:: enable_shared_global_addressbook
 - DAV:: get_calendar_contacts_config
 - DAV:: has_shared_global_addressbook
 - DAV:: is_dav_service_enabled

```

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',
    }
);

```

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

```

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

```

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

```

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

```

Notes:

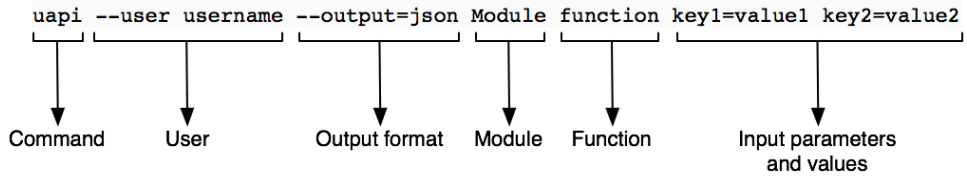
- UAPI calls via the command line do **not** return the metadata that other methods return if they experience errors that prevent a successful function run. For more information, read our [Guide to UAPI - Return Data](#) documentation.
- cPanel & WHM introduced the command line feature in version 56.

UAPI calls through the command line consist of the following basic parts:

Related Documentation

- [Use WHM API to Call cPanel API and UAPI](#) — You can call cPanel API and UAPI functions through the WHM API.
- [Guide to the Manage2 API](#) — The Manage2 API automates cPanel license management procedures.

- DAV:: is_horde_enabled
- ▼ DCV
 - DCV:: check_domains_via_dns
 - DCV:: check_domains_via_http
 - DCV:: ensure_domains_can_pass_dcv
- ▼ DomainInfo
 - DomainInfo:: domains_data
 - DomainInfo:: list_domains
 - DomainInfo:: main_domain_builtin_subdomain_aliases
 - DomainInfo:: single_domain_data
- ▼ DNS
 - DNS:: ensure_domains_reside_only_locally
 - DNS:: has_local_authority
- ▼ DNSSEC
 - DNSSEC:: disable_dnssec
 - DNSSEC:: enable_dnssec
 - DNSSEC:: fetch_ds_records
 - DNSSEC:: set_nsec3
 - DNSSEC:: unset_nsec3
- ▼ Email
 - Email:: account_name
 - Email:: add_auto_responder
 - Email:: add_domain_forwarder
 - Email:: add_forwarder
 - Email:: add_list
 - Email:: add_mailman_delegates
 - Email:: add_mx
 - Email:: add_pop
 - Email:: add_spam_filter



Part	Description
<i>Command</i>	<p>This value is always <code>uapi</code> for calls to UAPI.</p> <div style="border: 1px solid yellow; padding: 10px; margin: 10px 0;"> <p>Note: If you run CloudLinux™, you must use the full path of the <code>uapi</code> command:</p> <pre style="border: 1px dashed blue; padding: 5px; text-align: center;">/usr/local/cpanel/bin/uapi</pre> </div>
<i>User</i>	The account-level username.

cPanel interface's features.

- Guide to WHM API 0 — WHM API 0 accesses the WHM interface's features.
- Guide to cPanel API 2 — cPanel API 2 accesses cPanel account information and modifies settings.

Email:: browse_mailbox

Email:: change_mx

Email:: check_fastmail

Email:: check_pipe_forwarder

Email:: delete_auto_responder

Email:: delete_dominant_forwarder

Email:: delete_filter

Email:: delete_forwarder

Email:: delete_help_messages

Email:: delete_list

Email:: delete_mx

Email:: delete_pop

Email:: disable_filter

Email:: disable_mailbox_autocreate

Email:: disable_spam_assassin

Email:: disable_spam_autodelete

Email:: disable_spam_box

Email:: dispatch_client_settings

Email:: edit_pop_quota

Email:: enable_filter

Email:: enable_mailbox_autocreate

Email:: enable_spam_assassin

Email:: enable_spam_box

Email:: fetch_charmaps

Email:: fts_rescan_mailbox

Email:: generate_mailman_opt

Email:: get_auto_responder

Email:: get_chase

Output Type

The API output type that you wish to receive.

- Use `--output=json` to return JSON-formatted output.
- Use `--output=jsonpretty` to return indented JSON-formatted output.
- Use `--output=xml` to return XML-formatted output.

Important:
cPanel & WHM version 66 **deprecated** XML output.

- Use `--output=yaml` to return YAML-formatted output.

Note:
This parameter defaults to `--output=yaml`.

Module

The UAPI module name.

Function

The UAPI function.

User

The cPanel account username.

Input parameters and values

The function's input parameters and their values.

ts

Email: get_client_settings

Email: get_default_email_quota

Email: get_default_email_quota_mib

Email: get_disk_usage

Email: get_filter

Email: get_held_message_count

Email: get_mailbox_autocreate

Email: get_mailman_delegates

Email: get_main_account_disk_usage

Email: get_main_account_disk_usage_bytes

Email: get_max_email_quota

Email: get_max_email_quota_mib

Email: get_pop_quota

Email: get_spam_settings

Email: get_webmail_settings

Email: has_delegated_mailman_lists

Email: has_plaint_authentication

Email: hold_outgoing

Email: is_integer

Email: list_autoresponders

Email: list_default_address

Email: list_domain_forwarders

Email: list_filters

Email: list_filters_backups

Email: list_forwarders

Email: list_forward

- You **must** [URI-encode](#) values.
- Space-separate multiple parameter=value pairs.
- 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 examples:

```
uapi
--user
username
Module
function
key=["sslinstall\
","\video
tut\]"
uapi
--user
username
Module
function
key='{ "videotut", "
sslinstall"}'
```

ers_backups

Email:: list_lists

Email:: list_mail_domains

Email:: list_mx

Email:: list_pops

Email:: list_pops_with_disk

Email:: list_system_filter_info

Email:: passwd_list

Email:: passwd_pop

Email:: release_outgoing

Email:: remove_mailman_delegates

Email:: reorder_filters

Email:: set_always_accept

Email:: set_default_address

Email:: set_list_privacy_options

Email:: stats_db_status

Email:: store_filter

Email:: suspend_incoming

Email:: suspend_login

Email:: suspend_outgoing

Email:: trace_filter

Email:: unsuspend_incoming

Email:: unsuspend_login

Email:: unsuspend_outgoing

Email:: verify_password

▼ EmailAuth

EmailAuth:: disable_dkim

EmailAuth:: enable_dkim

EmailAuth:: ensure_dkim_keys_exist

Note:

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

Note:

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

```
uapi --help
```

Note:

For more information about browser-based UAPI calls, read the [cPanel or Webmail Session URL calls](#) section below.

This example uses the following variables:

Variable	Description	Possible values	Example
Module	The function's module.	Any UAPI module.	Email
function	The UAPI function.	Any UAPI function in the specified module.	addpop
parameters	The function's input parameters.	Any parameter from the specified function.	domain
value	The value to assign to the input parameter.	Parameters accept a variety of values.	example.com

cPanel or Webmail Session URL calls

You can access UAPI functions directly from any web browser when you make the call from a cPanel or Webmail session.

Browser-based calls consist of the following basic parts:

```
https://example.com:2083/cpsess#####/execute/Email/add_mx?domain=example.com&exchange=example.com
```

Server and port
Security token
Action
Module
Function
Input parameter and value

EmailAuth:: fetch_dkim_private_keys

EmailAuth:: install_dkim_private_keys

EmailAuth:: install_spf_records

EmailAuth:: validate_current_dkims

EmailAuth:: validate_current_ptrs

EmailAuth:: validate_current_spfs

ExternalAuthentication

ExternalAuthentication:: add_auth_link

ExternalAuthentication:: configured_modules

ExternalAuthentication:: get_auth_links

ExternalAuthentication:: has_external_auth_modules_configured

ExternalAuthentication:: remove_auth_link

Features

Features:: get_feature_metadata

Features:: has_feature

Features:: list_features

Fileman

Fileman:: autocompletedir

Fileman:: empty_trash

Fileman:: get_file_content

Fileman:: get_file_information

Fileman:: list_files

Fileman:: save_file_content

Fileman:: transcoding

Fileman:: upload_files

Part	Description
<i>Server and port</i>	<p>The web server's HTTP address and the port number to use. Generally, this value begins with <code>https://</code> and the domain name.</p> <p>When you call this API, use the following ports:</p> <ul style="list-style-type: none"> 2082 —Unsecure calls as a specific cPanel account. 2083 — Secure calls as a specific cPanel account. 2095 —Unsecure calls via a Webmail session. 2096 — Secure calls via a Webmail session. <div style="border: 1px solid red; padding: 5px; margin-top: 10px;"> <p>Warning: Unless you call this API via the WHM API, you cannot call this API via WHM's ports (2086 or 2087).</p> </div>
<i>Security token</i>	<p>The current session's security token.</p> <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p>Note: Cookie-based calls (for example, calls from a web browser) require a security token. For other authentication methods, read our Guide to API Authentication documentation.</p> </div>
<i>Action</i>	This portion of the URL will always be executed.
<i>Module</i>	The UAPI module name.
<i>Function</i>	The UAPI function.
<i>Input parameters and values</i>	<p>The UAPI function's input parameters and their values.</p> <ul style="list-style-type: none"> Separate multiple <code>parameter=value</code> pairs with the ampersand character (&). You must URI-encode these values. <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p>Note: The term "Boolean" in our documentation refers to parameters that accept values of 1 or 0. cPanel & WHM's APIs do not support the literal values of <code>true</code> and <code>false</code>.</p> </div>

- Ftp:: add_ftp
 - Ftp:: allows_anonymous_ftp
 - Ftp:: allows_anonymous_ftp_incoming
 - Ftp:: delete_ftp
 - Ftp:: ftp_exists
 - Ftp:: get_ftp_daemon_info
 - Ftp:: get_port
 - Ftp:: get_quota
 - Ftp:: get_welcome_message
 - Ftp:: kill_session
 - Ftp:: list_ftp
 - Ftp:: list_ftp_with_disk
 - Ftp:: list_sessions
 - Ftp:: passwd
 - Ftp:: server_name
 - Ftp:: set_anonymous_ftp
 - Ftp:: set_anonymous_ftp_incoming
 - Ftp:: set_homedir
 - Ftp:: set_quota
 - Ftp:: set_welcome_message
- Integration
 - Integration:: fetch_url
- KnownHosts
 - KnownHosts:: create
 - KnownHosts:: delete
 - KnownHosts:: update
 - KnownHosts:: verify
- LangPHP
 - LangPHP:: php_get_domain_handler
 - LangPHP:: php_get_impacted_domains
 - LangPHP:: php_get_installed_version

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.

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

▼ Command-line call...

s

LangPHP:: php_get_system_default_version

LangPHP:: php_get_vhost_versions

LangPHP:: php_ini_get_user_basic_directives

LangPHP:: php_ini_get_user_content

LangPHP:: php_ini_get_user_paths

LangPHP:: php_ini_set_user_basic_directives

LangPHP:: php_ini_set_user_content

LangPHP:: php_set_vhost_versions

▼ LastLogin

LastLogin:: get_last_or_current_logged_in_ip

▼ Locale

Locale:: get_attributes

▼ Mailboxes

Mailboxes:: expunge_mailbox_messages

Mailboxes:: expunge_messages_for_mailbox_guid

Mailboxes:: get_mailbox_status

Mailboxes:: has_utf8_mailbox_names

Mailboxes:: set_utf8_mailbox_names

▼ Market

Market:: cancel_pending_ssl_certificate

Market:: create_shopping_cart

Market:: get_all_products

Market:: get_certificate_status_details

Market:: get_login_url

Market:: get_pending_ssl_certificates

```
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/cpsess#####/execute/Module/function?parameter=value&parameter=value&parameter=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...

Market:: get_provider_specific_dcvc_constraints

Market:: get_providers_list

Market:: get_ssl_certificate_if_available

Market:: process_ssl_pending_queue

Market:: request_ssl_certificates

Market:: set_status_of_pending_queue_items

Market:: set_url_after_checkout

Market:: validate_login_token

▼ Mime

Mime:: add_handler

Mime:: add_hotlink

Mime:: add_mime

Mime:: add_redirect

Mime:: delete_handler

Mime:: delete_hotlink

Mime:: delete_mime

Mime:: delete_redirect

Mime:: list_handlers

Mime:: list_hotlinks

Mime:: list_mime

Mime:: list_redirects

Mime:: redirect_info

▼ ModSecurity

ModSecurity:: disable_all_domains

ModSecurity:: disable_domains

ModSecurity:: enable_all_domains

ModSecurity:: enable_domains

```
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/cpse  
ss#####/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
```

ModSecurity:: has
_modsecurity_inst
alled

ModSecurity:: list_
domains

▼ Mysql

Mysql:: add_host

Mysql:: add_host_
note

Mysql :: check_dat
abase

Mysql:: create_dat
abase

Mysql:: create_use
r

Mysql:: delete_dat
abase

Mysql:: delete_hos
t

Mysql:: delete_use
r

Mysql:: get_host_n
otes

Mysql:: get_privile
ges_on_database

Mysql:: get_restrict
ions

Mysql:: get_server
_information

Mysql:: locate_ser
ver

Mysql:: rename_d
atabase

Mysql:: rename_us
er

Mysql:: repair_dat
abase

Mysql:: revoke_ac
cess_to_database

Mysql:: set_passw
ord

Mysql:: set_privile
ges_on_database

▼ Notifications

Notifications:: get_
notifications_count

▼ NVData

NVData:: get

NVData:: set

▼ Parser

Parser:: firstfile_rel
ative_uri

▼ 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](#)

- PassengerApps:: disable_application
 - PassengerApps:: edit_application
 - PassengerApps:: enable_application
 - PassengerApps:: ensure_deps
 - PassengerApps:: list_applications
 - PassengerApps:: register_application
 - PassengerApps:: unregister_application
- PasswdStrength:: get_required_strength
- Personalization::set
- Personalization::get
- Postgresql:: create_database
 - Postgresql:: create_user
 - Postgresql:: delete_database
 - Postgresql:: get_restrictions
 - Postgresql:: grant_all_privileges
 - Postgresql:: rename_database
 - Postgresql:: rename_user
 - Postgresql:: rename_user_no_password
 - Postgresql:: revoke_all_privileges
 - Postgresql:: set_password
- Pushbullet:: send_test_message
- Quota:: get_quota_info

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/cpess#####/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...

```
cpapil --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...

- Resellers:: list_accounts
- ResourceUsage:: get_usages
- Restore:: directory_listing
 - Restore:: get_users
 - Restore:: query_file_info
 - Restore:: restore_file
- Session:: create_temp_user
- SiteTemplates:: list_site_templates
 - SiteTemplates:: list_user_settings
 - SiteTemplates:: publish
- SpamAssassin:: clear_spam_box
 - SpamAssassin:: get_symbolic_test_names
 - SpamAssassin:: get_user_preferences
 - SpamAssassin:: update_user_preference
- SSH:: get_port
- SSL:: add_autossl_excluded_domains
 - SSL:: can_ssl_redirect
 - SSL:: check_shared_cert
 - SSL:: delete_cert
 - SSL:: delete_csr
 - SSL:: delete_key
 - SSL:: delete_ssl
 - SSL:: disable_mail_sni

```
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](#).

SSL:: enable_mail
_sni

SSL:: fetch_best_f
or_domain

SSL:: fetch_cert_in
fo

SSL:: fetch_certific
ates_for_fqdns

SSL:: fetch_key_a
nd_cabundle_for_
certificate

SSL:: find_certifica
tes_for_key

SSL:: find_csrs_for
_key

SSL:: generate_ce
rt

SSL:: generate_cs
r

SSL:: generate_ke
y

SSL:: get_autossl_
excluded_domains

SSL:: get_autossl_
pending_queue

SSL:: get_autossl_
problems

SSL:: get_cabundl
e

SSL:: get_cn_nam
e

SSL:: install_ssl

SSL:: installed_ho
st

SSL:: installed_ho
sts

SSL:: is_autossl_c
heck_in_progress

SSL:: is_mail_sni_
supported

SSL:: is_sni_supp
orted

SSL:: list_certs

SSL:: list_csrs

SSL:: list_keys

SSL:: list_ssl_item
s

SSL:: mail_sni_sta
tus

SSL:: rebuild_mail
_sni_config

SSL:: rebuildssldb

SSL:: remove_aut
ossl_excluded_do
mains

SSL:: set_autossl_
excluded_domains

SSL:: set_cert_frie
ndly_name

SSL:: set_csr_frien
dly_name

SSL:: set_key_frie
ndly_name

SSL:: set_primary
_ssl

SSL:: set_ssl_shar
e

SSL:: show_cert

SSL:: show_csr

SSL:: show_key

SSL:: start_autossl_
_check

SSL:: toggle_ssl_r
edirects_for_doma
ins

SSL:: upload_cert

SSL:: upload_key

▼ StatsBar

StatsBar:: get_stat
s

▼ Styles

Styles:: current

Styles:: list

Styles:: set_default

Styles:: update

▼ SubDomain

SubDomain:: adds
ubdomain

▼ Themes

Themes:: get_the
me_base

Themes:: list

Themes:: update

▼ Tokens

Tokens:: create_fu
ll_access

Tokens:: list

Tokens:: rename

Tokens:: revoke

▼ TwoFactorAuth

TwoFactorAuth:: generate_user_configuration

TwoFactorAuth:: get_user_configuration

TwoFactorAuth:: remove_user_configuration

TwoFactorAuth:: set_user_configuration

▼ UserManager

UserManager:: check_account_conflicts

UserManager:: create_user

UserManager:: delete_user

UserManager:: dismiss_user

UserManager:: edit_user

UserManager:: list_users

UserManager:: lookup_service_account

UserManager:: lookup_user

UserManager:: merge_service_account

UserManager:: unlink_service_account

▼ UserTasks

UserTasks:: delete

UserTasks:: retrieve

▼ VersionControl

VersionControl:: create

VersionControl:: delete

VersionControl:: retrieve

VersionControl:: update

▼ VersionControlDeployment

VersionControlDeployment:: create

VersionControlDeployment:: delete

VersionControlDeployment:: retrieve

▼ WebDisk

WebDisk:: delete_user

WebDisk:: set_homedir

WebDisk:: set_password

WebDisk:: set_permissions

▼ WebmailApps

WebmailApps:: list_webmailapps

▼ WebVhosts

WebVhost:: list_domains

WebVhost:: list_ssl_capable_domains

▼ WordPressBackup

WordPressBackup:: any_running

WordPressBackup:: cancel

WordPressBackup:: cleanup

WordPressBackup:: get_available_backups

WordPressBackup:: is_running

WordPressBackup:: start

▼ WordPressInstance

Manager

WordPressInstanceManager:: cancel_scan

WordPressInstanceManager:: cleanup_scan

WordPressInstanceManager:: change_admin_password

WordPressInstanceManager:: configure_autoupdate

WordPressInstanceManager:: get_api_version

WordPressInstanceManager:: get_in

stance_by_id

WordPressInstanceManager::get_instances

WordPressInstanceManager::get_latest_wordpress_version_from_wordpress_org

WordPressInstanceManager::get_scan_results

WordPressInstanceManager::is_installable_addon_available

WordPressInstanceManager::start_scan

▼ WordPressRestore

WordPressRestore::any_running

WordPressRestore::cleanup

WordPressRestore::start