

# UAPI Functions - SSL::generate\_csr

## Guide to UAPI

[Return Data](#)

[Filter Output](#)

[Sort Output](#)

[Paginate Output](#)

[Output Columns](#)

[Calls from the WHM API](#)

[Custom UAPI Modules](#)

Backup:: [fullbackup\\_to\\_ftp](#)

Backup:: [fullbackup\\_to\\_homedir](#)

Backup:: [fullbackup\\_to\\_scp\\_with\\_key](#)

Backup:: [fullbackup\\_to\\_scp\\_with\\_password](#)

Backup:: [list\\_backups](#)

Bandwidth:: [get\\_enabled\\_protocols](#)

Bandwidth:: [get\\_retention\\_periods](#)

Bandwidth:: [query](#)

Batch:: [strict](#)

Brand:: [read](#)

Branding:: [file](#)

Branding:: [get\\_applications](#)

Branding:: [get\\_application\\_information](#)

Branding:: [get\\_available\\_applications](#)

Branding:: [get\\_information\\_for\\_applications](#)

Branding:: [image](#)

Branding:: [include](#)

Branding:: [spritelist](#)

CacheBuster:: [read](#)

CacheBuster:: [update](#)

Chkservd:: [get\\_exam\\_ports](#)

## Description

This function generates a certificate signing request (CSR).



### Note:

This function requires a valid key in the account's `ssl` directory. You can generate a key with UAPI's `SSL::generate_key` function.

## Examples

```
https://hostname.example.com:2083/cpsess#####/execute/SSL/generate_csr?domains=example.com&pass=12345luggage&countryName=US&stateOrProvinceName=Texas&localityName=Houston&organizationName=Organization&organizationalUnitName=Department&emailAddress=user%40example.com&key_id=example_com_a74f7_9dddf_1446659822_eae1ff43c4ec1991e195877c55755eff&friendly_name=TestCSR
```



### Note:

This example calls the UAPI function via a cPanel session. For more information, read our [Guide to UAPI](#) documentation.

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

// Generate a certificate signing request for the example.com site.
$SSL_generate_csr = $cpanel->uapi(
    'SSL', 'generate_csr', array(
        'key_id' =>
            'example_com_a74f7_9dddf_1446659822_eae1ff43c4ec1991e195877c55755eff',
        'pass' => '12345luggage',
        'domains' => 'example.com',
        'countryName' => 'US',
        'stateOrProvinceName' => 'Texas',
        'localityName' => 'Houston',
        'organizationName' => 'Organization',
        'organizationalUnitName' => 'Department',
        'emailAddress' => 'user@example.com',
        'friendly_name' => 'TestCert',
    )
);
```



### Note:

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

## Function information

### Module:

SSL  
API  
Version:

[UAPI](#)

### Available in:

cPanel 11.42+

### Methods:

GET, POST

### Required

### Parameters:

domains, countryName, stateOrProvinceName, localityName, organizationName, key\_id

### Return format:

JSON, XML

### Legacy Equivalents:

[SSL::gencsr](#)

- [SSL::gencsr](#)
- [SSL::gencsr](#)

## About this module

The SSL module manages certificates, certificate requests, and keys on your server.

## About UAPI

UAPI accesses and modifies data for cPanel accounts.



### Notes:

Chkserverd:: [get\\_exit\\_ports\\_ssl](#)

Chrome:: [get\\_dom](#)

Contactus:: [is\\_enabled](#)

cPAddons:: [get\\_available\\_addons](#)

cPAddons:: [get\\_instance\\_settings](#)

cPAddons:: [list\\_addon\\_instances](#)

cPGreyList:: [disable\\_all\\_domains](#)

cPGreyList:: [disable\\_domains](#)

cPGreyList:: [enable\\_all\\_domains](#)

cPGreyList:: [enable\\_domains](#)

cPGreyList:: [has\\_greylisting\\_enabled](#)

cPGreyList:: [list\\_domains](#)

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](#)

DAV:: [is\\_horde\\_enabled](#)

DCV:: [check\\_domains\\_via\\_dns](#)

DCV:: [check\\_domains\\_via\\_http](#)

DCV:: [ensure\\_domains\\_can\\_pass\\_dcv](#)

DomainInfo:: [domains\\_data](#)

DomainInfo:: [list\\_domains](#)

DomainInfo:: [main\\_domain\\_built\\_in\\_subdomain\\_aliases](#)

DomainInfo:: [single\\_domain\\_data](#)

DNS:: [ensure\\_domains\\_reside\\_only\\_locally](#)

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

# Generate a certificate signing request for the example.
com site.
my $SSL_generate_csr = $cpliveapi->uapi(
    'SSL', 'generate_csr', {
        'key_id' =>
'example_com_a74f7_9ddf_1446659822_eae1ff43c4ec1991e19587
7c55755eff',
        'pass' => '12345luggage',
        'domains' => 'example.com',
        'countryName' => 'US',
        'stateOrProvinceName' => 'Texas',
        'localityName' => 'Houston',
        'organizationName' => 'Organization',
        'organizationalUnitName' => 'Department',
        'emailAddress' => 'user@example.
com',
        'friendly_name' => 'TestCert',
    }
);
```



**Note:**

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

```
<!-- Generate a certificate signing request for the
example.com site.-->
[% data = execute( 'SSL', 'generate_csr', { pass =>
'12345luggage', friendly_name => 'TestCSR', domains =>
'example.com', countryName => 'US', stateOrProvinceName'
=> 'Texas', localityName => 'Houston', organizationName
=> 'Organization', organizationalUnitName =>
'Department', emailAddress => 'user@example.com', key_id
=>
'example_com_a74f7_9ddf_1446659822_eae1ff43c4ec1991e19587
7c55755eff', friendly_name => 'TestCert',} ); %]
[% FOREACH q = data %]
    <p>
        [% q %]
    </p>
[% END %]
```



**Note:**

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

• These functions may require specific configuration. To enable or disable features, use WHM's [Feature Manager](#) interface. • Unless you call this API, the WHM

DNS:: [has\\_local\\_authority](#)

DNSSEC:: [disable\\_dnssec](#)

DNSSEC:: [enable\\_dnssec](#)

DNSSEC:: [fetch\\_ds\\_records](#)

DNSSEC:: [set\\_nsec3](#)

DNSSEC:: [unset\\_nsec3](#)

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](#)

Email:: [browse\\_mailbox](#)

Email:: [change\\_mx](#)

Email:: [check\\_fastmail](#)

Email:: [check\\_pipe\\_forwarder](#)

Email:: [delete\\_auto\\_responder](#)

Email:: [delete\\_domain\\_forwarder](#)

Email:: [delete\\_filter](#)

Email:: [delete\\_forwarder](#)

Email:: [delete\\_held\\_messages](#)

Email:: [delete\\_list](#)

Email:: [delete\\_mx](#)

Email:: [delete\\_pop](#)

Email:: [disable\\_filter](#)

Email:: [disable\\_mailbox\\_autocreate](#)

Email:: [disable\\_spam\\_assassin](#)

```
uapi --user=username SSL generate_csr domains=example.com
pass=12345luggage countryName=US
stateOrProvinceName=Texas localityName=Houston
organizationName=Organization
organizationalUnitName=Department emailAddress=user%
40example.com
key_id=example_com_a74f7_9dddf_1446659822_eae1ff43c4ec1991
e195877c55755eff friendly_name=TestCSR
```



#### Notes:

- You **must** [URI-encode](#) values.
- `username` represents your account-level username.
- For more information and additional output options, read our [Guide to UAPI](#) documentation or run the `uapi --help` command.
- If you run CloudLinux™, you **must** use the full path of the `uapi` command:

```
/usr/local/cpanel/bin/uapi
```

M  
A  
P  
I  
,  
y  
o  
u  
m  
u  
s  
t  
u  
s  
e  
t  
h  
e  
a  
p  
p  
r  
o  
p  
r  
i  
a  
t  
e  
c  
P  
a  
n  
e  
l  
p  
o  
r  
t  
s  
(  
2  
0  
8  
2  
o  
r  
2  
0  
8  
3  
).  
Y  
o  
u  
c  
a  
n  
a  
l  
s  
o  
c  
a  
l  
l  
s  
o  
m  
e  
f  
u  
n  
c  
t  
i  
o  
n  
s  
v  
i  
a  
c  
P  
a  
n  
e  
l  
'  
s  
W  
e  
b  
m  
a  
i  
l  
p  
o  
r  
t  
s  
(  
2  
0  
9  
5  
o  
r  
2  
0  
9  
6  
).

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\\_charsets](#)

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](#)

```
{
  "data": {
    "commonName": "example.com",
    "created": "1366386445",
    "domains": [
      "example.com"
    ],
    "friendly_name": "example.com",
    "id":
      "example_com_bbe7e_16e2d_a626d1b34b62cefb13076a25b6831ca5"
  },
  "modulus": "THE MODULUS",
  "text": "-----BEGIN CERTIFICATE REQUEST-----
MIIC3TCCAcUCAQAwGzcxFDASBgNVBAMC2V4YW1wbGUuY29tMRYwFAYDVQQLDA1E
b2N1bWVudGF0aW9uMQswCQYDVQQGEwJVUzEiMCAGCSqGSIb3DQEJARYTBG
FlcmVudF1cmVudF1cmVudF1cmVudF1cmVudF1cmVudF1cmVudF1cmVudF1
Y2VAY3BhbWVsLm5ldDEUMBIGA1UECgwLY1BhbWVsIEluYy4xNDJAMBGNVBA
gMBVR1
eGFzMRAdDgYDVQQHDAdlb3VzdG9uMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ
8AMIIB
CgKCAQEAp092NvpqyT7mg2qDqZW1XetQGeo217o9r7A0ONaTrNsMgbl
/d31rQzpc
D65SPxJ/NSpovMi48tbue3A5W6+r2CN62oFk
/d34N2swN1w11NTLzOfTPQgGnBe0
CRdqNonx6818Cwj+auRMGakqxtaESofx0T87Ngr
/1CIqUbTzU0LembWh+1VI5Djy
2CRmRWKkSAAu7X
/OJN7cfcPh7ReZc9Ha6VmzNxbutBDRcnCJVOAGMsv2Cg4mQZhg
yMC8KFlr/ktHneDdGJdyprmdr0qkqzMwTaYIb6M
/93Yp+bswGH4oyqa4GYeTpRu
200CeydxONOnYRZwFL0SnL2Gnvnd3wIDAQABoAAwDQYJKoZIhvcNAQEFBQ
ADggEB
AF1o1zJbcviWcdZUv+PJ7G8cAqFAXa8a3W2a0Y1U0BaMrpDfu0HpQgOtO6
2rr+i+
hkrIReuk7Isyl6GoZkX17KhQ1Vo9aW8a8Ur7+0RGDbxR1K9jJWnvSFn6cM
MqRSfA
DXpptxhqBFZJIZMc3pPX9Jqycuu8bxelVUKeSMe3Jxcl1ZXEXX47FeRzvv
d+WXF9
4Xu4KZXHdzb5tB6jPhzIaN0/3d92N
/KhvCcnw3YC2fEGDkp01Zaoa7072+aq5K3b
M1htFv9deAHqIHefV41iOrxCUCIKhdPIjvAm7741DHLzPpJVrsho
/NpA74LEkny5
FMwKKx+6Jgx1HSWucn5kWeK=
-----END CERTIFICATE REQUEST-----\n"
  },
  "errors": null,
  "messages": [
    "Certificate Signing Request generated!"
  ],
  "status": 1
}
```



#### Note:

Use cPanel's *API Shell* interface (*cPanel >> Home >> Advanced >> API Shell*) to directly test cPanel API calls.

## Parameters

Parameter	Type	Description	Possible values	Example
-----------	------	-------------	-----------------	---------

## Find a function

## Related functions

- [Tutorial - Call UAPI's SSL::install\\_ssl Function in Custom Code](#) — This tutorial creates custom code to call UAPI's `SSL::install_ssl` function.
- [UAPI Modules - Market](#) — The Market module manages connections to cPanel Market providers.
- [UAPI Functions - Market::cancel\\_pending\\_ssl\\_certificate](#) — This function cancels an order and removes the polling for a pending certificate.
- [UAPI Functions - Market::create\\_shopping\\_cart](#) — This function creates a shopping cart with which the system sends an order to the cPanel Store.
- [UAPI Functions - Market::get\\_all\\_products](#) — This function lists all products available through all enabled cPanel Market providers.

Email: [get\\_webmail\\_settings](#)

Email: [has\\_delegated\\_mailman\\_lists](#)

Email: [has\\_plaintext\\_authentication](#)

Email: [hold\\_outgoing](#)

Email: [is\\_integer](#)

Email: [list\\_auto\\_responders](#)

Email: [list\\_default\\_address](#)

Email: [list\\_domain\\_forwarders](#)

Email: [list\\_filters](#)

Email: [list\\_filters\\_backups](#)

Email: [list\\_forwarders](#)

Email: [list\\_forwarders\\_backups](#)

Email: [list\\_lists](#)

Email: [list\\_mail\\_domains](#)

Email: [list\\_mx](#)

Email: [list\\_pops](#)

Email: [list\\_pops\\_wtih\\_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\\_log\\_in](#)

domains	<i>string</i>	<b>Required</b> The domain(s) for which to generate the certificate.	A valid string. <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"><b>Note:</b> If there is more than one domain, this parameter <b>must</b> be a comma-separated list.</div>	example.com
countryName	<i>string</i>	<b>Required</b> The two-letter country code.	A valid <a href="#">two-letter ISO-3166</a> country code.	US
stateOrProvinceName	<i>string</i>	<b>Required</b> The state's or locality's full name.	A valid string.	Texas
localityName	<i>string</i>	<b>Required</b> The certificate's city name.	A valid string.	Houston
organizationName	<i>string</i>	<b>Required</b> The certificate's organization.	A valid string.	Organization
organizationalUnitName	<i>string</i>	The certificate's organizational unit or department name.  This parameter defaults to an empty string.	A valid string.	Department
emailAddress	<i>string</i>	A valid email address to associate with the certificate.  This parameter defaults to an empty string.	A valid email address.	user@example.com
key_id	<i>string</i>	<b>Required</b> The key's ID.	A valid string.	<div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">example_com _cb497_a394 d_139724967 1_d1272da8f 13a1fd83749 3a5ad1f0a0f3</div>
friendly_name	<i>string</i>	A friendly name for the new certificate.  This parameter defaults to the domain name for which you generated the certificate.	A valid string.	TestCert

## Returns

Return	Type	Description	Possible values	Example
--------	------	-------------	-----------------	---------

Email:: [suspend\\_outgoing](#)

Email:: [trace\\_filter](#)

Email:: [unsuspend\\_incoming](#)

Email:: [unsuspend\\_login](#)

Email:: [unsuspend\\_outgoing](#)

Email:: [verify\\_password](#)

EmailAuth:: [disable\\_dkim](#)

EmailAuth:: [enable\\_dkim](#)

EmailAuth:: [ensure\\_dkim\\_keys\\_exist](#)

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:: [add\\_authn\\_link](#)

ExternalAuthentication:: [configured\\_modules](#)

ExternalAuthentication:: [get\\_authn\\_links](#)

ExternalAuthentication:: [has\\_external\\_auth\\_modules\\_configured](#)

ExternalAuthentication:: [remove\\_authn\\_link](#)

Features:: [get\\_feature\\_metadata](#)

Features:: [has\\_feature](#)

Features:: [list\\_features](#)

Fileman:: [autocomplete\\_dir](#)

Fileman:: [empty\\_trash](#)

Fileman:: [get\\_file\\_content](#)

modulus	<i>string</i>	The CSR's modulus.	A valid string.	a74f7636fa6ac93ee6836a83a995b55deb5019ea36d7ba3dafb00e38d693acdb0c81b2ff777d6b433a420fae523f127f352a68bcc8b8f2d6ee7b70395bafabd8237ada8164fd8df8376b30375c35d4d4cbce7d33d08069c17b409176a3689f1ebc97c0b08fe6ae44c19a92ac6d68448e7f1d13f3b360afd4222a51b4f35342de99b5a1fb5548e438f2d824664562a448002eed7fce24dedc7dc3e1ed179973d1dae959b33716eeb410d172708954e00632cbf60a0e26419860c8c0bc285d6bfe4b479de0dd189772a5dae676bd2a92acc0c1369821be8cfd88a7e6ecc061f8a32a9ae0661e4e946ed8e3827b277138d3a76116567cbd129cbd869ef9dddf
created	<i>string</i>	When the CSR was created.	A valid date in Unix time format.	1366386445
domains	<i>string</i>	The domains that the CSR covers.	A valid domain or domains.	example.com
commonName	<i>string</i>	The name that issued the CSR.	Usually, a valid domain.	example.com
friendly_name	<i>string</i>	The CSR's friendly name.	A valid string.	TestCSR
id	<i>string</i>	The CSR's ID.	A valid string.	example_com_a74f7_9dddf_1446659822_eae1ff43c4ec1991e195877c55755eff
text	<i>string</i>	The CSR's text.	A valid CSR.	-----BEGIN CERTIFICATE REQUEST----- MIIC3TCCAcUCAQAwgZcxFDASBGNVBAMMC2V4YW1wbGUuY29tMRYwFAYDVQQLEDA1b2N1bWVudGF0aW9uMQswCQYDVQQGEwJVUzEIMCAGCSqGSIb3DQEJARYTbGFMVUuY2VAY3BhbWVsLm5ldDEUMBIGA1UECgwLY1BhbmVsIEUuYy4xDjAMBGNVBAgMBVR1eGFzMRAwDgYDVQQHDAdlb3VzdG9uMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBBgcKCAQEAp092NvpqyT7mg2qDqZw1XetQGeo217o9r7A0ONaTrNsMgbl/d31rQzpcD65SPxJ /NSpovMi48tbue3A5W6+r2CN62oFk/d34N2swN1w1NLTzOfTPQgGnBe0CRdqNonx6818Cwj+auRMGakqxtaESofx0T87Ngr/1CIqUbtZU0LembWh+1VI5Djy2CRMWRKkSAAu7X /OJN7cfcPh7ReZc9Ha6VmzNxbutBDRcnCJV OAGMsv2Cg4mQZhg yMC8KF1r /ktHneDdGJdyprmdr0qkqzMwTaYIb6M/93Yp+bswGH4oyqa4GYeTpRu200CeydxONOnYRZWfL0SnL2Gnvnd3wIDAQABoAAwDQYJKoZIhvcNAQEFBQADggEB AF1oLzJbcviWcdZUv+PJ7G8cAqFAXa8a3W2a0Y1U0BaMrpDfuOHpQgOtO62rr+i+hkrIREuk7Isy16GoZkX17KhQ1V09aW8a8Ur7+ORGDbxRlK9jJWnvSFn6cMMqRSFA DXppt.xhqBFZJIZMc3pPX9Jqycuu8bxIvUK eSM3Jxc11ZXEXX47FeRZxvd+WXF94Xu4KZXHdz5tB6jPhzIaN0/3d92N /KhvCcnw3YC2FEGDkpOLZaoa7072+aq5K3b MlhtFv9deAHqIHeFv41iOrxUCIKIhdPIjvA m774LDHLzPpJvrsho/NpA74LEkny5 FMwKKx+6Jgx1HSWucn5kWek=-----END CERTIFICATE REQUEST-----

Fileman:: [get\\_file\\_information](#)

Fileman:: [list\\_files](#)

Fileman:: [save\\_file\\_content](#)

Fileman:: [transcode](#)

Fileman:: [upload\\_files](#)

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:: [fetch\\_url](#)

KnownHosts:: [create](#)

KnownHosts:: [delete](#)

KnownHosts:: [update](#)

KnownHosts:: [verify](#)

LangPHP:: [php\\_get\\_domain\\_handler](#)

LangPHP:: [php\\_get\\_impacted\\_domains](#)

LangPHP:: [php\\_get\\_installed\\_versions](#)

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:: [get\\_last\\_or\\_current\\_logged\\_in\\_ip](#)

Locale:: [get\\_attributes](#)

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:: [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](#)

Market:: [get\\_provider\\_specific\\_dcv\\_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:: 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:: disable\_all\_domains

ModSecurity:: disable\_domains

ModSecurity:: enable\_all\_domains

ModSecurity:: enable\_domains

ModSecurity:: has\_modsecurity\_installed

ModSecurity:: list\_domains

Mysql:: add\_host

Mysql:: add\_host\_note

Mysql:: check\_database

Mysql:: create\_database

Mysql:: create\_user

Mysql:: delete\_database

Mysql:: delete\_host

Mysql:: delete\_user

Mysql:: get\_host\_notes

Mysql:: get\_privileges\_on\_database

Mysql:: get\_restrictions

Mysql:: get\_server\_information

Mysql:: locate\_server

Mysql:: rename\_database

Mysql:: rename\_user

Mysql:: repair\_database

Mysql:: revoke\_access\_to\_database

Mysql:: set\_password

Mysql:: set\_privileges\_on\_database

Notifications:: get\_notifications\_count

NVData:: get

NVData:: set

Parser:: firstfile\_relative\_uri

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](#)

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  
SSL:: enable\_mail\_sni  
SSL:: fetch\_best\_for\_domain  
SSL:: fetch\_cert\_info  
SSL:: fetch\_certificates\_for\_fqdns  
SSL:: fetch\_key\_and\_cabundle\_for\_certificate  
SSL:: find\_certificates\_for\_key  
SSL:: find\_csrs\_for\_key  
SSL:: generate\_cert  
SSL:: generate\_csr  
SSL:: generate\_key  
SSL:: get\_autossl\_excluded\_domains  
SSL:: get\_autossl\_pending\_queue  
SSL:: get\_autossl\_problems  
SSL:: get\_cabundle  
SSL:: get\_cn\_name  
SSL:: install\_ssl  
SSL:: installed\_host  
SSL:: installed\_hosts  
SSL:: is\_autossl\_check\_in\_progress  
SSL:: is\_mail\_sni\_supported  
SSL:: is\_sni\_supported  
SSL:: list\_certs  
SSL:: list\_csrs  
SSL:: list\_keys  
SSL:: list\_ssl\_items  
SSL:: mail\_sni\_status  
SSL:: rebuild\_mail\_sni\_config

SSL:: rebuildssldb

SSL:: remove\_autossl\_excluded\_domains

SSL:: set\_autossl\_excluded\_domains

SSL:: set\_cert\_friendly\_name

SSL:: set\_csr\_friendly\_name

SSL:: set\_key\_friendly\_name

SSL:: set\_primary\_ssl

SSL:: set\_ssl\_share

SSL:: show\_cert

SSL:: show\_csr

SSL:: show\_key

SSL:: start\_autossl\_check

SSL:: toggle\_ssl\_rejects\_for\_domains

SSL:: upload\_cert

SSL:: upload\_key

StatsBar:: get\_stats

Styles:: current

Styles:: list

Styles:: set\_default

Styles:: update

SubDomain:: addsubdomain

Themes:: get\_theme\_base

Themes:: list

Themes:: update

Tokens:: create\_full\_access

Tokens:: list

Tokens:: rename

Tokens:: revoke

TwoFactorAuth:: generate\_user\_configuration

TwoFactorAuth:: get\_user\_configuration

TwoFactorAuth:: remove\_user\_configuration

TwoFactorAuth:: [set\\_user\\_configuration](#)

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:: [delete](#)

UserTasks:: [retrieve](#)

VersionControl:: [create](#)

VersionControl:: [delete](#)

VersionControl:: [retrieve](#)

VersionControl:: [update](#)

VersionControlDeployment:: [create](#)

VersionControlDeployment:: [delete](#)

VersionControlDeployment:: [retrieve](#)

WebDisk:: [delete\\_user](#)

WebDisk:: [set\\_home\\_dir](#)

WebDisk:: [set\\_password](#)

WebDisk:: [set\\_permissions](#)

WebmailApps:: [list\\_emailapps](#)

WebVhost:: [list\\_domains](#)

WebVhost:: [list\\_ssl\\_capable\\_domains](#)

WordPressBackup::  
[any\\_running](#)

WordPressBackup::  
[cancel](#)

WordPressBackup::  
[cleanup](#)

WordPressBackup::  
[get\\_available\\_backups](#)

WordPressBackup::  
[is\\_running](#)

WordPressBackup::  
[start](#)

WordPressInstance  
Manager::[cancel\\_scan](#)

WordPressInstance  
Manager::[cleanup\\_scan](#)

WordPressInstance  
Manager::[change\\_admin\\_password](#)

WordPressInstance  
Manager::[configure\\_autoupdate](#)

WordPressInstance  
Manager::[get\\_api\\_version](#)

WordPressInstance  
Manager::[get\\_instance\\_by\\_id](#)

WordPressInstance  
Manager::[get\\_instances](#)

WordPressInstance  
Manager::[get\\_latest\\_wordpress\\_version\\_from\\_wordpress\\_org](#)

WordPressInstance  
Manager::[get\\_scan\\_results](#)

WordPressInstance  
Manager::[is\\_installable\\_addon\\_available](#)

WordPressInstance  
Manager::[start\\_scan](#)

WordPressRestore::  
[any\\_running](#)

WordPressRestore::  
[cleanup](#)

WordPressRestore::  
[start](#)

