

Guide to API Authentication - Secure Remote Logins

Guide to API Authentication

[Access Hash Authentication](#)

[API Tokens](#)

[Browser-Based Authentication](#)

[Secure Remote Logins](#)

[Single Sign On](#)

[Two-Factor Authentication](#)

[Username and Password Authentication](#)

Introduction

cPanel & WHM supports secure remote logins via the `Cpanel::LogMeIn` module.

Important:

- API calls that use a method that includes a URL **must** use the correct port:
 - 2082 — Unsecure calls to cPanel's APIs.
 - 2083 — Secure calls to cPanel's APIs.
 - 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.
 - 2095 — Unsecure calls to cPanel's APIs via a Webmail session.
 - 2096 — Secure calls to cPanel's APIs via a Webmail session.
- Otherwise-correct calls will return `Permission denied` or `Function not found` errors if they use an incorrect port number.
- This document **only** includes cPanel & WHM authentication methods. For Manage2 authentication information, read our [Guide to the Manage2 API](#) documentation.

Related documentation

Secure remote logins

To allow users to log in to cPanel & WHM remotely, write a script that calls the `Cpanel::LogMeIn` module's `get_loggedin_url()` subroutine. This subroutine returns a URL for remote logins to cPanel, WHM, or Webmail.

Parameters

When you call the `get_loggedin_url()` subroutine, you **must** pass in the following parameters:

Parameter	Type	Description	Possible values	Example
<code>user</code>	<i>string</i>	Required The cPanel, WHM, or Webmail username.	A valid username. In most scripts, retrieve this value from user input.	<code>username</code>
<code>pass</code>	<i>string</i>	Required The account's password.	A secure password. In most scripts, retrieve this value from user input.	<code>123456luggage</code>
<code>hostname</code>	<i>string</i>	Required The cPanel, WHM, or Webmail account's main domain.	A valid domain on the server.	<code>example.com</code>
<code>service</code>	<i>string</i>	Required The account's service type.	<ul style="list-style-type: none"><code>cpanel</code><code>whm</code><code>webmail</code>	<code>cpanel</code>

goto_uri

string

Required

The location to which the script will redirect the user. The get_loggedin_url () subroutine uses this value to create the URL.

A valid path, relative to the /usr/local/cpanel/base/directory.

/frontend/per_lantern/park/index.html

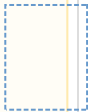
Note

- For the best system speed and performance, we recommend that you append the /json-app

Content by Label

There is no content with the specified labels

i / d u m m y s t r i n g t o y o u r g o t o - u r i p a r a m e t e r . F o r e x a m p l e

			<p>e :</p>  <ul style="list-style-type: none"> • W e i n t r o d u c e d t h i s f u n c t i o n a l i t y i n c P a n e l & W H M v e r s i o n 7 6 .
--	--	--	---

Returns

The `get_loggedin_url()` subroutine returns the following values:

Return	Type	Description	Possible values	Example
<code>login_ok</code>	<i>boolean</i>	Whether the function authenticated the user successfully.	<ul style="list-style-type: none"> • 1 — Success. • 0 — Failure. 	1

login_message	string	A status message from the subroutine.	<ul style="list-style-type: none"> • Login OK • An error message. 	Login OK
login_url	string	The secure remote login URL.	A valid URL that includes a valid session token and directs the user to the goto_uri path.	https://example.com:2083/cpsess#####/frontend/paper_lantern/park/index.html

Example Perl script

[Click to view...](#)

```
#!/usr/bin/perl

use lib '/usr/local/cpanel';
use Cpanel::LogMeIn ();

# The cPanel, WHM, or Webmail
# username entered at login.
my $user = 'YYYYYY';

# The cPanel, WHM, or Webmail
# password entered at login.
my $pass = 'XXXXXX';

# The cPanel, WHM, or Webmail
# account's main domain.
my $host = 'localhost';

# The service to log in to (cpanel,
# whm, or webmail).
my $service = 'cpanel';

# Call the get_loggedin_url
# subroutine.
my ( $login_ok, $login_message,
    $login_url, $security_token ) =
    Cpanel::LogMeIn::get_loggedin_url(
        'user' => $user,
        'pass' => $pass,
        'hostname' => $host,
        'service' => $service,
        'goto_uri' => '/'
    );

# Set up HTTP headers with the
# returned values.
if ($login_ok) {
    print "Location:
    $login_url\r\n\r\n";
}
else {
```

```
        print "Content-type:
text/plain\r\n\r\n";
        print "LOGIN FAILED:
$login_message\n";
    }

# Call the get_loggedin_url
subroutine for a different URL.
( $login_ok, $login_message,
$login_url, $security_token ) =
Cpanel::LogMeIn::get_loggedin_url(
    'active_token_for_reuse' =>
$security_token,

    'user'      => $user,
    'pass'      => $pass,
    'hostname'  => $host,
    'service'   => $service,
    'goto_uri'  =>
'/frontend/paper_lantern/park/index.h
tml'
);

# Set up HTTP headers with the
returned values.
if ($login_ok) {
    print "Location:
$login_url\r\n\r\n";
}
else {
    print "Content-type:
text/plain\r\n\r\n";
    print "LOGIN FAILED:
$login_message\n";
}
```

}