



# InstaMember USER'S GUIDE

## InstaMember Licensing API Guide

## InstaMember Licensing API Guide

The InstaMember licensing feature is designed to integrate seamlessly with your web applications or scripts. It gives you the ability, not only to protect your web apps/scripts, but also to track who is using your apps/scripts, and where they are using it.

You can use this licensing feature to protect your web Scripts, such as WordPress plugins/themes, or any regular web applications/scripts written in PHP, ASP, Java, etc. In fact, the licensing feature can protect any type of applications/software as long as the programming language that you use to build your software have the ability to communicate with a REST API server via remote HTTP call, and can parse a JSON encoded message.

# TABLE OF CONTENTS

<b>Title Page</b>	<b>1</b>
<b>Inroductory Page</b>	<b>2</b>
<b>Table of Contents</b>	<b>3</b>
<b>How It Works</b>	<b>4</b>
<b>Setting Up Your Licensing Server</b>	<b>5</b>
<b>How To Enable The Licensing Feature For Your Product</b>	<b>6</b>
<b>PHP Quick-Code Samples</b>	<b>7</b>
<b>License Keys Management</b>	<b>11</b>
<b>Reference</b>	<b>12</b>
<b>Credits &amp; Contact Page</b>	<b>14</b>

## How It Works

This is a quick explanation of how the InstaMember Licensing feature can protect your product.

### Step #1: Check License State During Initialization

So, your users installed your script on their server, and they open it up. During initialization of your script, do a check whether the script has been licensed or not. If yes, then you can run and display your script as it supposed to be.

But if it hasn't been licensed, then do NOT run and display the main user-interface of your script yet, but instead, display a form where your software users can enter and submit their license key (auto-generated by InstaMember during purchase) to unlock the script. At this point, users cannot do anything with the script, but they must activate it first.

### Step #2: License Activation

Once your users entered their license key into the form and click submit, your script needs to make an API call to your own licensing server to check if the license is valid or not. If it's valid, then you can make a "secret" entry into the database or a file so your script can be known that it has been activated, and then unlock your script so your users can start using your product. But if it's not valid, then you give the users an error message, and do not unlock your script until the users enter a valid license key.

## Setting Up Your Licensing Server

Each time you installed InstaMember on a web server (under WordPress), you actually installed a licensing server too. This means your InstaMember installation will also act as the licensing server, and that's where InstaMember will keep the master data of your customers' license keys and its usage.

What your script needs to do is to call your own licensing server to validate and activate your customers' license key via a remote HTTP call. But before you can make any call, you must obtain the API URI and the Secret word, which is unique for every InstaMember installation. Here's how to that:

- In your "InstaMember" admin panel, go to the "Setup/Configuration -> General."
- Scroll down to find the "Licensing Server Configuration" section, and click the "+" button to open up its settings.
- You'll find two fields under the "Licensing Server Configuration." The first one is "Licensing Server Secret" and the other is "Licensing Server URL."
- The "Licensing Server Secret" will be used for authentication when an API Call is made. By default, InstaMember have already generated a random key for this setting, however, you can change it if you wish.
- The "Licensing Server URL" is your license API URI.
- Your API URI will look like this:  
`http:// YOURDOMAIN.COM /!/instamember_api/license`

If you installed WP in a sub-directory, then the API URI will look like this:  
`http://YOURDOMAIN.COM/WPDIR!/instamember_api/license`

- Click the "Update Settings" button to save any changes.

You can use the InstaMember's licensing API in two different ways. The first is for software activation, and the other is to validate the user's license key periodically.

Every API request requires basic HTTP authentication, and support both "POST" and "GET" methods. The responses will be returned in the JSON serialization format.

## How To Enable The Licensing Feature For Your Product

In order to use the InstaMember's licensing feature, you must first enable it in your InstaMember installation. Each of the products that has been created in InstaMember have its own licensing settings.

Here's how to enable the licensing feature for one of your products:

- Login to your WordPress Dashboard as an administrator.
- Navigate to the InstaMember admin panel, and click the "Products" tab.
- Edit the product that will be protected by InstaMember license, and click on the "Licensing" sub-menu.
- Enable the product licensing feature.
- You can set the license limitation on the "Usage Per License" setting. For example, if you want to limit a license so it can be used on a single domain only, then enter '1' and choose "Domain Name" from the drop down. Or, you can simply choose 'No Limit' for unlimited use.
- Optionally, you can set the 'Prefix' and 'Suffix' for the license key. The 'prefix' and 'suffix' will be included in the generated license key. Both 'prefix' and 'suffix' values are static, so it will be the same for each customer.
- The default length of the generated license is 10 characters. Please note that the length of 'prefix' and 'suffix' will not be counted. So, if the combined 'prefix' and 'suffix' length is 8, and the license's length is 12 then the total length of the generated license key for each customer will be 20 characters.
- You can also set the license key expiration in the "Expire In" setting.
- Click the "Save License Settings" button when you're done.

## PHP Quick-Code Samples

### Code Sample To Activate a New License Key

**Description:** *Remote call to validate a license key and also log/write the user's data into the database. You have to use this method upon software activation so InstaMember can count or limit the usage properly, and also to log it into the database.*

HTTP Method: *POST*

API End-Point:

`http://YOURDOMAIN.COM/!/instamember_api/license/{license_key}`

#### PHP Example:

```
<?php
$secret_key = 'YOUR SECRET KEY';
$license_key = 'LICENSE KEY HERE';
$md5_hash = md5($license_key . $secret_key);
$api_url =
"http://yourdomain.com/!/instamember_api/license/{$license_key}";
$hostname = @gethostbyaddr($_SERVER['SERVER_ADDR']);
$data = array(
'domain' => $_SERVER['HTTP_HOST'],
'userip' => $_SERVER['REMOTE_ADDR'],
'servip' => ( $hostname ) ? $hostname : $_SERVER['SERVER_ADDR']
);
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $api_url);
curl_setopt($ch, CURLOPT_USERPWD, "{$secret_key}:{$md5_hash}");
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
curl_setopt($ch, CURLOPT_TIMEOUT, 10);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($data));
$result = curl_exec($ch);
curl_close($ch);
$response = json_decode($result);
If ( $response->status == 'Success' ) {
    // code to unlock software here...
} else {
```

```
} // code handling error here...  
}
```

## Code Sample to Check If a License Key is Still Valid, or Has Been Expired, Or Suspended

**Description:** Remote call to check if a license key is still active or invalid (e.g. expired, inactive, suspended, etc). This method does not accept any type of data, and will not log any data into the database.

HTTP Method: *GET*

API End-Point:

`http://YOURDOMAIN.COM/!/instamember_api/license/{license_key}`

### PHP Example:

```
<?php  
$secret_key = 'YOUR SECRET KEY';  
$license_key = 'LICENSE KEY HERE';  
$md5_hash = md5($license_key . $secret_key);  
$api_url =  
"http://yourdomain.com/!/instamember_api/license/{$license_key}";  
$ch = curl_init();  
curl_setopt($ch, CURLOPT_URL, $api_url);  
curl_setopt($ch, CURLOPT_USERPWD, "{$secret_key}:{$md5_hash}");  
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);  
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);  
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);  
curl_setopt($ch, CURLOPT_TIMEOUT, 10);  
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);  
$result = curl_exec($ch);  
curl_close($ch);  
  
$response = json_decode($result);  
If ( $response->status == 'Success' ) {  
    // do nothing... simply let your users to continue using your software  
} else if ( $response->status == 'Expired' ) {  
    // the license key is expired... do something here...  
} else if ( $response->status == 'Suspended' ) {  
    // the license key has been suspended... do something here...  
} else {  
    // do something here...
```



```
}
```

## Code Sample to Get a License Key Detail

**Description:** Remote call to get a license key detail from the licensing server database, such as the associated product ID and the license settings for that product.

HTTP Method: *GET*

API End-Point:

[http://YOURDOMAIN.COM/!/instamember\\_api/license/{license\\_key}/data](http://YOURDOMAIN.COM/!/instamember_api/license/{license_key}/data)

### PHP Example:

```
<?php
$secret_key = 'YOUR SECRET KEY';
$license_key = 'LICENSE KEY HERE';
$md5_hash = md5($license_key . $secret_key);
$api_url =
"http://yourdomain.com/!/instamember_api/license/{${license_key}}";
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $api_url);
curl_setopt($ch, CURLOPT_USERPWD, "{$secret_key}:{${md5_hash}}");
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
curl_setopt($ch, CURLOPT_TIMEOUT, 10);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$result = curl_exec($ch);
curl_close($ch);
$response = json_decode($result);
echo '<pre>' . print_r($response, true) . '</pre>';
```

**When you run the above code correctly, you should get something like this:**

```
stdClass Object
(
    [status] => Success
    [message] => License key is active
    [data] => stdClass Object
    (
        [ID] => 22
```

```
[license] => 'LICENSE KEY HERE',
[product_id] => 3
[tx_id] => 73
[member_id] => 36
    [status] => 1
    [date] => 1364211772
    [expiration] => 1 year
    [limit] => 3 domains
    [created] => 2013-03-25 11:42:52
    [expired] => 1395747772
    [expired_date] => 2014-03-25 11:42:52
)
)
```

## More Code Samples

We've also created detail code samples so that you can learn further on how to use this licensing feature on real-life PHP applications.

There are two code samples. The first one is an example on how to apply the licensing feature on a regular PHP script. The other is a code sample to integrate InstaMember's licensing feature with a WordPress plugin.

Both of the sample codes can be downloaded below:

<http://insta-member.com/IMLicenseEx.zip>

<http://insta-member.com/IMSamplePlugin.zip>

## License Keys Management

To check and maintain the license key that has been generated by InstaMember, simply login to your WordPress Dashboard and navigate to **"InstaMember -> Manage Licenses"**.

You can view license keys that have been generated along with the details, including the member data, associated product and also the usage log. You can also "deactivate" or "reactivate" or "delete" a license key from this page.

## Reference

### Return Status

**Success** – License key is active or valid

**Maxed** – License key is active but has reached the maximum usage limit (returned only upon activation)

**Expired** – License key has expired

**Inactive** – License key is not active or disabled by admin

**Suspended** – License key has been suspended by admin

**Unauthorized** – Authentication request failed

**Error** – Any type of errors

### Return Data Example (in JSON)

License Active/Success:

```
{
  "status": "Success",
  "message": "License key is active"
}
```

License Inactive:

```
{
  "status": "Inactive",
  "message": "License key is inactive or disabled"
}
```

License Expired:

```
{
  "status": "Expired",
  "message": "License key has been expired"
}
```

License Maxed:

```
{
  "status": "Maxed",
  "message": "License key is active but has reached the maximum usage
limit"
}
```

License Suspended:

```
{
  "status": "Suspended",
  "message": "License key has been suspended"
}
```

Authentication Failed:

```
{
```

```
    "status": "Unauthorized",
    "message": "Authentication failed"
}
License key is empty:
{
    "status": "Error",
    "message": "License key is empty"
}
License is not found in the database:
{
    "status": "Error",
    "message": "Invalid license key"
}
No post data upon activation:
{
    "status": "Error",
    "message": "Required fields are empty"
}
```

## **BROUGHT TO YOU BY: SUZANNA THERESIA**

If you have any questions or input on how we could improve this manual, please contact our support team at <http://asksuzannatheresia.com>