



# JSON API USER PLUS PLUGIN VERSION 1.4.0

API Documentation

---

*JSON API User Plus extends the JSON API Plugin to allow RESTful user registration, authentication and many other User Meta, BuddyPress functions.*

---

Ali Qureshi  
support@wppim.com  
parorrey.com  
March 21, 2015

## Contents

|       |                                    |   |
|-------|------------------------------------|---|
| 1     | Overview .....                     | 5 |
| 1.1   | WordPress Versions .....           | 5 |
| 2     | Plugin Settings Page .....         | 5 |
| 2.1   | API Key .....                      | 5 |
| 2.2   | Disable Nonce .....                | 5 |
| 3     | Documentation .....                | 6 |
| 3.1   | Method: get_nonce .....            | 6 |
| 3.1.1 | Example .....                      | 6 |
| 3.1.2 | How to get nonce .....             | 6 |
| 3.2   | Method: info .....                 | 6 |
| 3.2.1 | Required argument .....            | 6 |
| 3.2.2 | Example .....                      | 6 |
| 3.3   | Method: register .....             | 6 |
| 3.3.1 | Required arguments .....           | 6 |
| 3.3.2 | Optional arguments .....           | 6 |
| 3.3.3 | Example .....                      | 6 |
| 3.4   | Method: fb_connect .....           | 7 |
| 3.4.1 | Required argument .....            | 7 |
| 3.4.2 | Example .....                      | 7 |
| 3.5   | Method: email_exists .....         | 7 |
| 3.5.1 | Required argument .....            | 7 |
| 3.5.2 | Example .....                      | 7 |
| 3.6   | Method: username_exists .....      | 7 |
| 3.6.1 | Required argument .....            | 7 |
| 3.6.2 | Example .....                      | 7 |
| 3.7   | Method: validate_auth_cookie ..... | 7 |
| 3.7.1 | Required argument .....            | 7 |
| 3.7.2 | Example .....                      | 7 |
| 3.8   | Method: generate_auth_cookie ..... | 8 |
| 3.8.1 | Required arguments .....           | 8 |
| 3.8.2 | Optional argument .....            | 8 |
| 3.8.3 | Example .....                      | 8 |
| 3.9   | Method: new_post .....             | 8 |
| 3.9.1 | Required argument .....            | 8 |
| 3.9.2 | Optional arguments .....           | 8 |

|        |                                 |    |
|--------|---------------------------------|----|
| 3.9.3  | Example.....                    | 8  |
| 3.10   | Method: update_post.....        | 9  |
| 3.10.1 | Required argument .....         | 9  |
| 3.10.2 | Optional arguments .....        | 9  |
| 3.10.3 | Example.....                    | 9  |
| 3.11   | Method: delete_post .....       | 9  |
| 3.11.1 | Required argument .....         | 9  |
| 3.11.2 | Optional arguments .....        | 9  |
| 3.11.3 | Example.....                    | 9  |
| 3.12   | Method: get_user_meta .....     | 10 |
| 3.12.1 | Required argument .....         | 10 |
| 3.12.2 | Optional argument.....          | 10 |
| 3.13   | Method: update_user_meta.....   | 10 |
| 3.13.1 | Required arguments .....        | 10 |
| 3.13.2 | Example.....                    | 10 |
| 3.14   | Method: delete_user_meta.....   | 10 |
| 3.14.1 | Required arguments .....        | 10 |
| 3.14.2 | Example.....                    | 10 |
| 3.15   | Method: xprofile .....          | 10 |
| 3.15.1 | Required arguments .....        | 10 |
| 3.15.2 | Example.....                    | 10 |
| 3.16   | Method: xprofile_update.....    | 11 |
| 3.16.1 | Required arguments .....        | 11 |
| 3.16.2 | Example.....                    | 11 |
| 3.17   | Method: retrieve_password ..... | 11 |
| 3.17.1 | Required argument .....         | 11 |
| 3.17.2 | Example.....                    | 11 |
| 3.18   | Method: get_avatar .....        | 11 |
| 3.18.1 | Required argument .....         | 11 |
| 3.18.2 | Example.....                    | 11 |
| 3.19   | Method: get_userinfo .....      | 11 |
| 3.19.1 | Required argument .....         | 11 |
| 3.19.2 | Example.....                    | 11 |
| 3.20   | Method: post_comment.....       | 12 |
| 3.20.1 | Required arguments .....        | 12 |
| 3.20.2 | Example.....                    | 12 |

|        |                                 |    |
|--------|---------------------------------|----|
| 3.21   | Method: profile.....            | 12 |
| 3.21.1 | Required argument.....          | 12 |
| 3.21.2 | Example.....                    | 12 |
| 3.22   | Method: friends .....           | 12 |
| 3.22.1 | Required argument.....          | 12 |
| 3.22.2 | Example.....                    | 12 |
| 3.23   | Method: threads .....           | 12 |
| 3.23.1 | Required argument.....          | 12 |
| 3.23.2 | Optional arguments .....        | 12 |
| 3.23.3 | Example.....                    | 12 |
| 3.24   | Method: thread.....             | 13 |
| 3.24.1 | Required arguments .....        | 13 |
| 3.24.2 | Example.....                    | 13 |
| 3.25   | Method: new_message.....        | 13 |
| 3.25.1 | Required arguments .....        | 13 |
| 3.25.2 | Optional arguments .....        | 13 |
| 3.25.3 | Example.....                    | 13 |
| 3.26   | Method: mark_thread_read .....  | 13 |
| 3.26.1 | Required arguments .....        | 13 |
| 3.26.2 | Example.....                    | 13 |
| 3.27   | Method: mark_thread_unread..... | 13 |
| 3.27.1 | Required arguments .....        | 13 |
| 3.27.2 | Example.....                    | 13 |
| 3.28   | Method: messages_count.....     | 13 |
| 3.28.1 | Required arguments .....        | 13 |
| 3.28.2 | Example.....                    | 14 |
| 3.29   | Method: avatar_upload.....      | 14 |
| 3.29.1 | Required argument.....          | 14 |
| 3.29.2 | Optional arguments .....        | 14 |
| 3.29.3 | Example.....                    | 14 |
| 3.30   | Method: delete_account .....    | 14 |
| 3.30.1 | Required arguments .....        | 14 |
| 3.30.2 | Example.....                    | 14 |
| 4      | Changelog.....                  | 15 |
| 4.1    | Version 1.4 .....               | 15 |
| 4.2    | Version 1.3 .....               | 15 |

|     |                     |    |
|-----|---------------------|----|
| 4.3 | Version 1.2.0 ..... | 15 |
| 4.4 | Version 1.1.0 ..... | 15 |

# 1 Overview

**JSON API User Plus** extends the JSON API Plugin with a new Controller to allow RESTful user registration, authentication, password reset, RESTful Facebook Login, RESTful User Meta and BuddyPress xProfile get and update methods. **JSON API User Plus** plugin is for WordPress/Mobile app developers who want to use WordPress as mobile app data backend.

## 1.1 WordPress Versions

1. Requires at least: 3.0.1
2. Tested up to: 4.1.1
3. Lasted Updated: March 21, 2015

# 2 Plugin Settings Page

## 2.1 API Key

JSON API User Plus includes **API key variable** which protects and restricts the API endpoint calls. This key is available and can be updated from **Settings > User Plus** options page. Your app **must include this key with every call** to get the data from REST API.

Please see below for calling endpoints examples.

## 2.2 Disable Nonce

Nonce is required for some endpoints such as registration, add\_post, update\_post, delete\_account.

If you want to disable nonce check protection in some endpoints, such as registration, new\_post, update\_post, delete\_account, you can disable it in the plugin settings panel. **Settings > User Plus**

## 3 Documentation

### 3.1 Method: get\_nonce

nonce can be created by calling if you are registering user or use it in any endpoint where it is required. This method is from **JSON API Plugin** core controller.

#### 3.1.1 Example

`http://localhost/api/get_nonce/?controller=userplus&method=register`

You can then use 'nonce' value to register user.

#### 3.1.2 How to get nonce

`http://localhost/api/get_nonce/?controller=CONTROLLER-NAME-HERE&method=METHOD-NAME-HERE`

### 3.2 Method: info

#### 3.2.1 Required argument

- key

#### 3.2.2 Example

`http://localhost/api/userplus/info/?key=KEY-HERE`

This endpoint will return the plugin current version.

### 3.3 Method: register

#### 3.3.1 Required arguments

- username
- email
- nonce
- display\_name

#### 3.3.2 Optional arguments

- user\_pass
- notify – send 'no' value to disable registration notification to user

#### 3.3.3 Example

`http://localhost/api/userplus/register/?username=john&email=john@domain.com&nonce=8bdf4e16&display_name=John`

Optional params: 'user\_pass', 'notify', 'user\_nicename', 'user\_url', 'nickname', 'first\_name', 'last\_name', 'description', 'rich\_editing', 'user\_registered', 'role', 'jabber', 'aim', 'yim', 'comment\_shortcuts', 'admin\_color', 'use\_ssl', 'show\_admin\_bar\_front'.

**Please make sure you provide valid values that these fields expect in correct format.**

To disable registration email notification to user:

`http://localhost/api/user/register/?username=john&email=john@domain.com&nonce=8bdfef4e16&display_name=John&user_pass=8734tHYS&notify=no`

## 3.4 Method: fb\_connect

### 3.4.1 Required argument

- `access_token` - Provide valid `access_token` with email extended permission.

### 3.4.2 Example

`http://localhost/api/userplus/fb_connect/?key=KEY-HERE&access_token=CAACEdEose0cBADLKmcHWOZCnW4RGU8emG`

Provide valid `access_token` with email extended permission. To generate test `access_token`, try this tool <https://developers.facebook.com/tools/explorer/> and select the app from above drop down that you want to get `access_token` (You must have joined that app already with email permission to generate `access_token`) for and then select email from the fields. By default, only 'id' and 'name' are added but you need to include 'email' for user identification.

You will have to first allow extended permission for email in the app joining screen. Please note that above tool is only for testing, you generate valid `access_token` using the Graph API in your app. You should have prior knowledge of Facebook Graph API to use this endpoint.

## 3.5 Method: email\_exists

### 3.5.1 Required argument

- `email`

### 3.5.2 Example

`http://localhost/api/userplus/email_exists/?key=KEY-HERE&email=xyz@abc.com`

This endpoint will check if the given email address is available for registration or not.

## 3.6 Method: username\_exists

### 3.6.1 Required argument

- `username`

### 3.6.2 Example

`http://localhost/api/userplus/username_exists/?key=KEY-HERE&username=john`

This endpoint will check if the given username is available for registration or not.

## 3.7 Method: validate\_auth\_cookie

### 3.7.1 Required argument

- `cookie`

### 3.7.2 Example

`http://localhost/api/userplus/validate_auth_cookie/?key=KEY- HERE &cookie=admin|43089754375034fjwfn39u8`



## 3.8 Method: generate\_auth\_cookie

### 3.8.1 Required arguments

- username
- password

### 3.8.2 Optional argument

- seconds - 60 means 1 minute - default is for 14 days

### 3.8.3 Example

[http://localhost/api/userplus/generate\\_auth\\_cookie/?username=john&password=PASSWORD-HERE](http://localhost/api/userplus/generate_auth_cookie/?username=john&password=PASSWORD-HERE)

Optional 'seconds' var. If provided, generated cookie will be valid for that many seconds, otherwise default is for 14 days.

**Generate cookie for 1 minute:** [http://localhost/api/userplus/generate\\_auth\\_cookie/?username=john&password=PASSWORD-HERE&seconds=60](http://localhost/api/userplus/generate_auth_cookie/?username=john&password=PASSWORD-HERE&seconds=60)

## 3.9 Method: new\_post

### 3.9.1 Required argument

- cookie - for user authentication
- nonce - available from the `get_nonce` method (call with vars `controller=userplus` and `method=new_post`)
- status - sets the post status ("draft" or "publish")
- title - the post title
- content - the post content

### 3.9.2 Optional arguments

- categories - a comma-separated list of categories (URL slugs)
- tags - a comma-separated list of tags (URL slugs)
- attachment – upload field will cause an attachment to be stored with new post with image set as featured thumbnail

### 3.9.3 Example

URL to POST values: [http://localhost/api/userplus/new\\_post/](http://localhost/api/userplus/new_post/)

**Note:** User must have capability to add post. Change user role if needed.

## 3.10 Method: update\_post

### 3.10.1 Required argument

- `cookie` - for user authentication
- `post_id` – post ID to be updated
- `nonce` - available from the `get_nonce` method (call with vars `controller=userplus` and `method=update_post`)

### 3.10.2 Optional arguments

- `status` - sets the post status ("draft" or "publish")
- `title` - the post title
- `content` - the post content
- `categories` - a comma-separated list of categories (URL slugs)
- `tags` - a comma-separated list of tags (URL slugs)
- `post_media` – to delete all previously attached images from post, pass value 'delete'
- `attachment` – upload field will cause an attachment to be stored with new post with image set as featured thumbnail. Previously uploaded images will still be attached to post. Old media can be deleted by sending `post_media=delete`

### 3.10.3 Example

URL to POST values: `http://localhost/api/userplus/update_post/`

Note: User must have capability to update post. Change user role if needed.

## 3.11 Method: delete\_post

### 3.11.1 Required argument

- `cookie` - for user authentication
- `post_id` – post ID to delete
- `nonce` - available from the `get_nonce` method (call with vars `controller=userplus` and `method=delete_post`)

### 3.11.2 Optional arguments

- `post_media` – to delete all previously attached images from post, pass value 'delete'
- `force_delete` – if set to 'true', it will delete the post bypassing the WordPress trash.

### 3.11.3 Example

URL to POST values: `http://localhost/api/userplus/delete_post/`

Note: User must have capability to delete post. Change user role if needed.

## 3.12 Method: get\_user\_meta

### 3.12.1 Required argument

- user\_id

### 3.12.2 Optional argument

- meta\_key

`http://localhost/api/userplus/get_user_meta/?key=KEY-HERE&cookie=COOKIE-HERE&meta_key=KEY-HERE`

## 3.13 Method: update\_user\_meta

### 3.13.1 Required arguments

- cookie
- meta\_key
- meta\_value - You may send multiple values separated by comma.

### 3.13.2 Example

`http://localhost/api/userplus/update_user_meta/?key=KEY-HERE&cookie=COOKIE-HERE&meta_key=KEY-HERE&meta_value=VALUE-HERE`

## 3.14 Method: delete\_user\_meta

### 3.14.1 Required arguments

- cookie
- meta\_key
- meta\_value

### 3.14.2 Example

`http://localhost/api/userplus/delete_user_meta/?key=KEY-HERE&cookie=COOKIE-HERE&meta_key=KEY-HERE&meta_value=VALUE-HERE`

`http://localhost/api/userplus/update_user_meta/?key=KEY-HERE&cookie=COOKIE-HERE&&meta_key=KEY-HERE&meta_value=value1,value2,value3`

## 3.15 Method: xprofile

### 3.15.1 Required arguments

- user\_id
- any profile 'field' var – call by exact label

### 3.15.2 Example

`http://localhost/api/userplus/xprofile/?key=KEY-HERE&user_id=USERID-HERE&field=FIELD-LABEL-HERE`

## 3.16 Method: xprofile\_update

### 3.16.1 Required arguments

- cookie
- any profile 'field' var and 'value'

### 3.16.2 Example

`http://localhost/api/userplus/xprofile_update/?key=KEY-HERE&cookie=COOKIE-HERE&exact-xprofile-field-label=value`

`http://localhost/api/userplus/xprofile_update/?key=KEY-HERE&cookie=COOKIE-HERE&field=value&field2=value&multi-value-field=value1,value2,value3`

Please make sure you provide ending comma for all those fields which have multiple values. e.g. If 'skills' xProfile field has multiple values, pass them like

`http://localhost/api/userplus/xprofile_update/?key=KEY-HERE&cookie=COOKIE-HERE&skills=PHP,MySQL, or &skills=PHP,`

**Please make sure you always pass ending comma for multi-select fields to be added in array format.**

## 3.17 Method: retrieve\_password

### 3.17.1 Required argument

- user\_login

### 3.17.2 Example

`http://localhost/api/userplus/retrieve_password/?key=KEY-HERE&user_login=john`

## 3.18 Method: get\_avatar

### 3.18.1 Required argument

- user\_id

### 3.18.2 Example

`http://localhost/api/userplus/get_avatar/?key=KEY-HERE&user_id=1`

## 3.19 Method: get\_userinfo

### 3.19.1 Required argument

- user\_id

### 3.19.2 Example

`http://localhost/api/userplus/get_userinfo/?key=KEY-HERE&user_id=1`

## 3.20 Method: post\_comment

### 3.20.1 Required arguments

- cookie
- post\_id
- content
- comment\_status

### 3.20.2 Example

`http://localhost/api/userplus/post_comment/?key=KEY-HERE&cookie=COOKIE-HERE&post_id=ID&content=Comment contents here&comment_status=1`

## 3.21 Method: profile

### 3.21.1 Required argument

- user\_id

### 3.21.2 Example

`http://localhost/api/userplus/profile/?key=KEY-HERE&user_id=1`

## 3.22 Method: friends

### 3.22.1 Required argument

- cookie

### 3.22.2 Example

`http://localhost/api/userplus/friends/?key=KEY-HERE&cookie=COOKIE-HERE`

## 3.23 Method: threads

### 3.23.1 Required argument

- cookie

### 3.23.2 Optional arguments

- box - possible values are 'inbox', 'sent' (default is 'inbox')
- page - returns a specific page number from the paginated results (default is 1)
- per\_page - determines how many threads per page are returned (default value is 10)
- tsort - determines the threads sorting, possible values are 'asc' or 'desc' (default is 'desc')
- search - allows to search the term in the threads
- sender\_id - restricts the results to this sender\_id (this is beta feature)

### 3.23.3 Example

`http://localhost/api/userplus/threads/?key=KEY-HERE&cookie=COOKIE-HERE`

## 3.24 Method: thread

### 3.24.1 Required arguments

- cookie
- thread\_id

### 3.24.2 Example

<http://localhost/api/userplus/thread/?key=KEY-HERE&cookie=COOKIE-HERE>

## 3.25 Method: new\_message

### 3.25.1 Required arguments

- cookie
- recipient\_ids – can be more than one id. Multiple recipient\_ids must be comma separated
- subject
- content

### 3.25.2 Optional arguments

- thread\_id – to reply existing thread message, include 'thread\_id' variable, otherwise new thread will be started.

### 3.25.3 Example

[http://localhost/api/userplus/new\\_message/?key=KEY-HERE&cookie=COOKIE-HERE&recipient\\_ids=COMMA-SEPARATED-IDS&subject=Test Msg&content=Content is here](http://localhost/api/userplus/new_message/?key=KEY-HERE&cookie=COOKIE-HERE&recipient_ids=COMMA-SEPARATED-IDS&subject=Test Msg&content=Content is here)

## 3.26 Method: mark\_thread\_read

### 3.26.1 Required arguments

- Cookie
- thread\_id

### 3.26.2 Example

[http://localhost/api/userplus/mark\\_thread\\_read/?key=KEY-HERE&cookie=COOKIE-HERE&thread\\_id=12](http://localhost/api/userplus/mark_thread_read/?key=KEY-HERE&cookie=COOKIE-HERE&thread_id=12)

## 3.27 Method: mark\_thread\_unread

### 3.27.1 Required arguments

- Cookie
- thread\_id

### 3.27.2 Example

[http://localhost/api/userplus/mark\\_thread\\_unread/?key=KEY-HERE&cookie=COOKIE-HERE&thread\\_id=12](http://localhost/api/userplus/mark_thread_unread/?key=KEY-HERE&cookie=COOKIE-HERE&thread_id=12)

## 3.28 Method: messages\_count

### 3.28.1 Required arguments

- cookie

### 3.28.2 Example

[http://localhost/api/userplus/messages\\_count/?key=KEY-HERE&cookie=COOKIE-HERE](http://localhost/api/userplus/messages_count/?key=KEY-HERE&cookie=COOKIE-HERE)

## 3.29 Method: avatar\_upload

### 3.29.1 Required argument

- cookie
- avatar – file upload type var

### 3.29.2 Optional arguments

- full – BuddyPress avatar width and height (default is BP\_AVATAR\_FULL\_WIDTH or 300).
- thumb – BuddyPress avatar width and height (default is BP\_AVATAR\_THUMB\_WIDTH or 100).

### 3.29.3 Example

[http://localhost/api/userplus/avatar\\_upload/](http://localhost/api/userplus/avatar_upload/)

You must POST the form with avatar image in file upload field var with other required and optional variables as hidden fields.

## 3.30 Method: delete\_account

### 3.30.1 Required arguments

- cookie
- nonce – use get\_nonce for controller= userplus & method=delete\_account

### 3.30.2 Example

[http://localhost/api/userplus/delete\\_account/?key=KEY-HERE&cookie=COOKIE-HERE&nonce=NONCE-HERE](http://localhost/api/userplus/delete_account/?key=KEY-HERE&cookie=COOKIE-HERE&nonce=NONCE-HERE)

You can get nonce value to be used with this endpoint by going to

[http://localhost/api/get\\_nonce/?controller= userplus&method=delete\\_account](http://localhost/api/get_nonce/?controller= userplus&method=delete_account)

**Be very careful using 'delete\_account', this will delete the user account.**

## 4 Changelog

### 4.1 Version 1.4

1. Added 'Disable nonce' settings option to disable nonce check for endpoints where it was previously required.

### 4.2 Version 1.3

2. Added 'info' endpoint to show version of plugin.
3. Updated 'delete\_account' to show error message for super admin user cant be removed.
4. Updated 'register' endpoint to add default role for the registering user.

### 4.3 Version 1.2.0

1. Added 'new\_post', 'update\_post', 'delete\_post' endpoints.
2. Updated 'generate\_auth\_cookie' to include 'cookie\_name' value in the response.

### 4.4 Version 1.1.0

1. Added the function to authenticate, allow the user (with edit rights) to use JSON API core controllers as well.
2. Updated `generate\_auth\_cookie` to allow setting up auth cookie for any required duration. Just provide the `seconds` var with `nonce`, `username` and `password` to get required cookie. Default time is 14 days.
3. Updated `register` to allow disabling notification email to user after registration. Provide `notify` var with value 'no' while registering and it won't send email. You must provide `user\_pass` var (which is optional otherwise) to use this because password is sent in the email. Without `user\_pass` var, this won't make sense disabling notification so it won't.
4. Fixed typos in documentation and added new documentation.