



WooCommerce REST API Integration

October 27, 2018



Andrew Duncan

CEO/Owner/Developer
Databuzz



The eCommerce platform for WordPress

The world's most customisable eCommerce platform

The most popular eCommerce platform on the web - over 47m downloads

AUTOMATIC

Part of the Automattic family

My WooCommerce API Journey



Could use SQL . . .

```
select
    p.order_id,
    p.order_item_id,
    p.order_item_name,
    p.order_item_type,
    max(CASE WHEN pm.meta_key = '_product_id' THEN pm.meta_value END ) as productID,
    CAST(max( CASE WHEN pm.meta_key = '_qty' THEN pm.meta_value END ) AS UNSIGNED) as Qty,
    max( CASE WHEN pm.meta_key = '_variation_id' THEN pm.meta_value END ) as variationID,
    CAST(max( CASE WHEN pm.meta_key = '_line_total' THEN pm.meta_value END ) AS UNSIGNED) as lineTotal,
    CAST(max( CASE WHEN pm.meta_key = '_line_subtotal_tax' THEN pm.meta_value END ) AS UNSIGNED) as subTotalTax,
    CAST(max( CASE WHEN pm.meta_key = '_line_tax' THEN pm.meta_value END ) AS UNSIGNED) as Tax,
    max( CASE WHEN pm.meta_key = '_tax_class' THEN pm.meta_value END ) as taxClass,
    CAST(max( CASE WHEN pm.meta_key = '_line_subtotal' THEN pm.meta_value END ) AS UNSIGNED) as subtotal

from ocm_woocommerce_order_items p join
    ocm_woocommerce_order_itemmeta pm
    on p.order_item_id = pm.order_item_id
where order_item_type = 'line_item'
group by p.order_item_id
```

What is an API?

- an **application programming interface** (API) is a set of subroutine definitions, protocols, and tools for building software.
- it is a set of clearly defined methods of **communication** between various components
- if you want to get your software application talking to someone else's software/service their API tells you how to do it



Why use the WooCommerce API?

Integration with . . .

- your business Customer Relationship Management software (CRM)

- your accounting software (MYOB, Xero, SAP etc)

- remove any double handling/data entry

Update WooCommerce from . . .

- upload Products from your CRM/Inventory system to WooCommerce

- bulk updates: change categories, prices, descriptions etc in one operation

Create an Order and send link to Payment Page . . .

Benefits



WooCommerce REST API

`woocommerce.github.io/woocommerce-rest-api-docs`

v3 is the current version (requires WooCommerce 3.5.x or later and WordPress 4.4+)

v2 is the current version (requires WooCommerce 3.0.x or later and WordPress 4.4+)

v1 and previous legacy APIs now deprecated

REST API allows WooCommerce data to be created, read, updated, and deleted using requests in JSON format and using WordPress REST API Authentication methods and standard HTTP verbs which are understood by most HTTP clients

GET/POST/PUT/DELETE

Code samples for cURL, Node.js, PHP, Python and Ruby

API Authentication

Authentication over HTTP

for HTTP you can use OAuth 1.0a “one-legged” authentication but NOT recommended (use SSL/HTTPS)

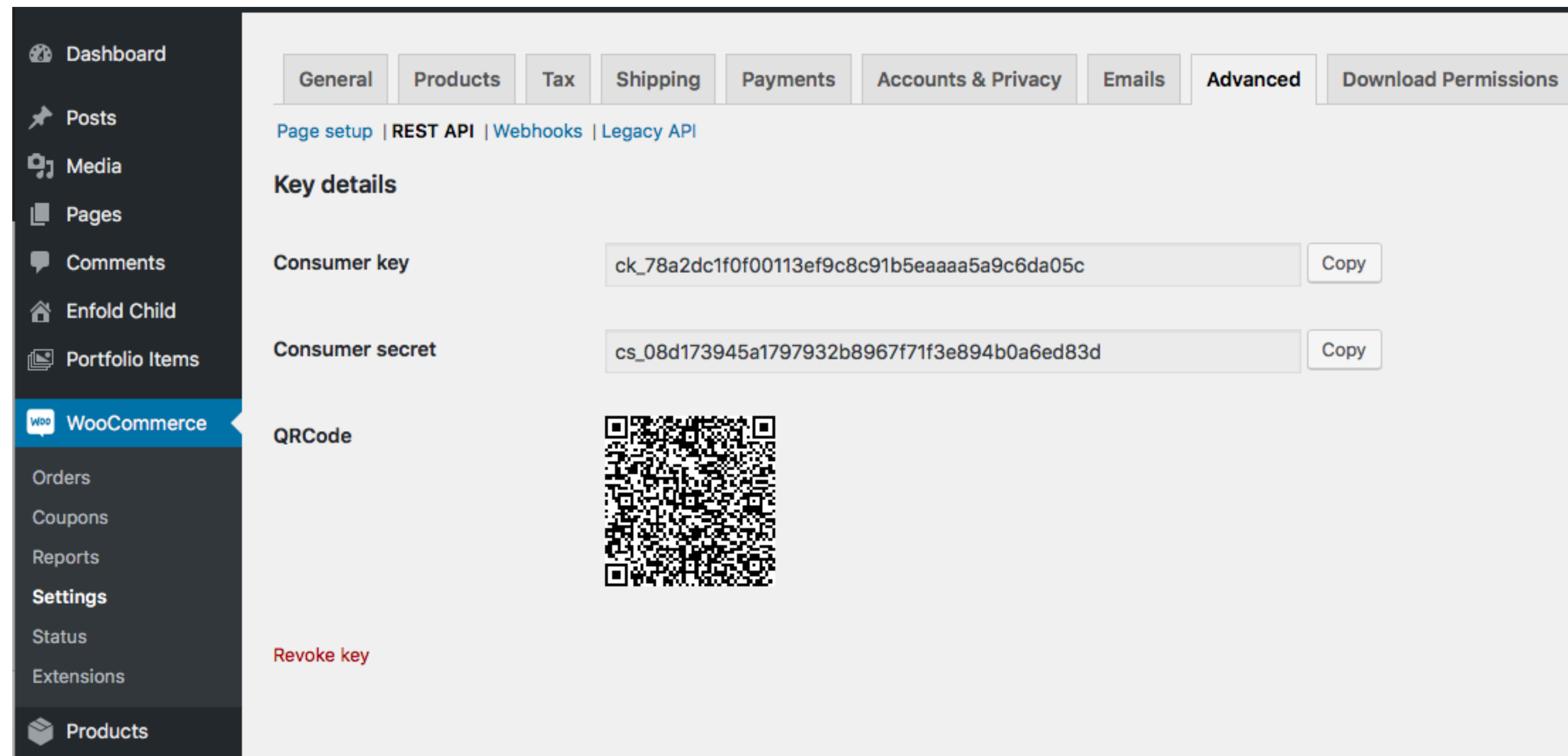
Authentication over HTTPS

use HTTP Basic Auth by providing the REST API Consumer Key as the username and the REST API Consumer Secret as the password

```
curl https://www.example.com/wp-json/wc/v3/orders  
-u consumer_key:consumer_secret
```


REST API Keys

generated either through the WordPress admin interface or they can be auto-generated through an endpoint



Create Request

POST to endpoint

/wp-json/wc/v3/customers

Authentication Header

"Content-Type: application/json" Header

Data JSON

Response will be JSON and include WooCommerce ID

Store the ID for subsequent updates/downloads/deletes

```
{ "id": 25 }
```

Create Request Example

```
curl -X POST https://example.com/wp-json/wc/v3/customers
  -u consumer_key:consumer_secret
  -H "Content-Type: application/json"
  -d '{
    "email": "john.doe@example.com",
    "first_name": "John",
    "last_name": "Doe",
    "username": "john.doe"
  }'
```

Update Request

PUT to endpoint

/wp-json/wc/v3/customers/**<id>**

Authentication Header

"Content-Type: application/json" Header

Data JSON (can send changes only)

Update Request Example

```
curl -X PUT https://example.com/wp-json/wc/v3/customers/25
  -u consumer_key:consumer_secret
  -H "Content-Type: application/json"
  -d '{
    "first_name": "James",
    "billing": {
      "first_name": "James"
    },
    "shipping": {
      "first_name": "James"
    }
  }'
```

Get Request - Single Record

GET to endpoint

`/wp-json/wc/v3/customers/<id>`

```
curl https://example.com/wp-json/wc/v3/customers/25  
-u consumer_key:consumer_secret
```

Response will be JSON (same as POST)

Get Request - Response

```
{  
  "id": 25,  
  "date_created": "2017-03-21T16:09:28",  
  "date_created_gmt": "2017-03-21T19:09:28",  
  "email": "john.doe@example.com",  
  "first_name": "John",  
  "last_name": "Doe",  
  "role": "customer",  
  "username": "john.doe"  
  ...  
}
```

Get Request - Multiple Records

GET to endpoint

`/wp-json/wc/v3/orders`

```
curl https://example.com/wp-json/wc/v3/orders
```

```
-u consumer_key:consumer_secret
```

Response will be JSON array

Can use parameters and pagination:

```
wp-json/wc/v3/orders?
```

```
after=2018-07-01T00:00:00Z&before=2018-07-31T23:59:59Z&page=1&orderby=id&order=asc
```

Pagination

Requests that return multiple items will be paginated to 10 items by default

- the items per page can be specified with the **?per_page** parameter

You can specify further pages with the **?page** parameter

The total number of resources and pages are always included in the **X-WP-Total** and **X-WP-TotalPages** HTTP Response Headers

Pagination info is included in the Link Header: next, last, first, prev

- use these values instead of building your own URLs

Response Headers

HTTP/1.1 200 OK

X-Powered-By: PHP/7.1.19

Content-Type: application/json; charset=UTF-8

Access-Control-Expose-Headers: X-WP-Total, X-WP-TotalPages

Access-Control-Allow-Headers: Authorization, Content-Type

X-WP-Total: 21

X-WP-TotalPages: 3

Link: <<https://sydneyfilemaker.com/wp-json/wc/v3/orders?page=2&orderby=id&order=asc&dp=2>>; rel="next"

Allow: GET, POST

Transfer-Encoding: chunked

Date: Sat, 21 Jul 2018 03:09:24 GMT

Accept-Ranges: bytes

Server: LiteSpeed

Alt-Svc: quic=":443"; ma=2592000; v="35,37,38,39"

Connection: Keep-Alive

Delete Request

DELETE to endpoint

/wp-json/wc/v3/customers/**<id>**

```
curl -X DELETE https://example.com/wp-json/wc/v3/customers/25?force=true  
-u consumer_key:consumer_secret
```

Some resources don't support trashing and require **force=true** parameter (check docs)

Response will be JSON

Webhooks

Send notifications to your server/CRM etc when certain actions occur

New Order, New Customer etc

Webhook data	
Name	<input type="text" value="New Order Webhook"/>
Status	<input type="text" value="Active"/>
Topic	<input type="text" value="Order created"/>
Delivery URL	<input type="text" value="https://fm01.databuzz.com.au/demos/WooCommerce/web"/>
Secret	<input type="text" value="j.DHAJA Y(Pau.MnPAvWoR*bY7 eWvQK_+w}.G nWw l6+9#"/>
API Version	<input type="text" value="WP REST API Integration v2"/>

Webhook Workflow Example



New Order Webhook



Get Order API Request



Upload to Xero via Xero API

Tips

Timestamps are ISO8601 format: `2017-03-22T16:28:02`

Check HTTP Response Codes:

- 200 and 201 for success
- 4xx for request errors (400, 401 etc)
- 500 server error etc

Upload Images via URL or WordPress Media ID (can use WordPress Media API endpoint to upload directly)

Custom Fields usually associated with Meta Data

Reports any bugs via GitHub (fixed quickly in my experience)

Use Postman to test requests (getpostman.com)

REST API v3

A **reviews** endpoint to list, get, create, and update product reviews.

Date fields are now editable on the **products** endpoint, and the endpoint has been updated to work with the “on backorder” stock status.

A series of new **reports** endpoints to get total counts of orders, products, customers, coupons and reviews.

New **orderby** options for the **products** endpoint to allow listing products by price, popularity, or rating.

Automatic total calculations when applying/removing coupons in the **orders** endpoint.

New **data** endpoints to fetch Continents, Countries and Currencies.

Refund line items through the API.

Ability to batch update **settings**

Demos

Q&A



www.databuzz.com.au



andrew@databuzz.com.au



[@aduncan_au](https://twitter.com/aduncan_au)



[databuzz.au](https://facebook.com/databuzz.au)

