



Plugin Directory

Changeset 3440173 for wordpress-simple-paypal-shopping-cart

Timestamp: 01/15/2026 09:16:28 AM (3 months ago)

Author: mra13

- Message:
- Improved the Stripe post-checkout handling code to prevent a potential issue with some server configurations.
 - Added output escaping to the 'wpsc_display_product' shortcode parameters.

Location: [wordpress-simple-paypal-shopping-cart/trunk](#)

Files: ■ 7 edited

- [includes/admin/wp_shopping_cart_menu_stripe_settings.php](#) (1 diff)
- [includes/wpsc-misc-checkout-ajax-handler.php](#) (2 diffs)
- [includes/wpsc-misc-functions.php](#) (2 diffs)
- [readme.txt](#) (2 diffs)
- [stripe.php](#) (6 diffs)
- [wp_shopping_cart.php](#) (3 diffs)
- [wp_shopping_cart_shortcodes.php](#) (2 diffs)

Unmodified Added Removed

wordpress-simple-paypal-shopping-cart/trunk/includes/admin/wp_shopping_cart_menu_stripe_settings.php Tabular | Unified

r3417800r3440173

154	154	<code><input type="checkbox" name="wpsc_auto_send_receipt_and_invoices" value="1" <?php esc_attr_e(\$wpsc_auto_send_receipt_and_invoices);?> /></code>
155	155	<code><p class="description"></code>
156		<code><?php _e("When enabled the receipt or invoice generated by stripe will automatically be sent to the customer.", "wordpress-simple-paypal-shopping-cart");?></code>
157		<code>
</code>
158		<code><?php echo sprintf(__("First you will need to select %s in stripe dashboard %s.", "wordpress-simple-paypal-shopping-cart"), 'Successful payments', 'Customer emails settings') ?></code>
	156	<code><?php echo sprintf(__("Complete the steps detailed in %s to activate the automated receipt feature.", "wordpress-simple-paypal-shopping-cart"), 'this guide') ?></code>
159	157	<code></p></code>
160	158	<code></td></code>

wordpress-simple-paypal-shopping-cart/trunk/includes/wpsc-misc-checkout-ajax-handler.php Tabular | Unified

r3417800r3440173

71	71	<code>}</code>
72	72	<code>}</code>
73		<code>\$query_args = array('simple_cart_stripe_ipn' => '1', 'ref_id' => \$wpsc_cart->get_cart_id());</code>
	73	<code>\$client_reference_id = \$cart_id; // TODO: old code. need to remove</code>
	74	<code>\$csid = "{CHECKOUT_SESSION_ID}"; // NOTE: Stripe replaces the {CHECKOUT_SESSION_ID} with actual session id before redirecting to this url.</code>
	75	

```

r3417800r3440173
76     $query_args = array( 'simple_cart_stripe_ipn' => '1', 'ref_id' => $client_reference_id, 'csid'
77     => $csid );
74     77     $stripe_ipn_url = add_query_arg( $query_args, WP_CART_SITE_URL );
75     78
...     ...
82     85
83     86     $opts = array(
84     'client_reference_id' => $cart_id,
85     87     'client_reference_id' => $client_reference_id,
86     88     'billing_address_collection' => $force_collect_address ? 'required' : 'auto',
87     89     'mode' => 'payment',
88     90     'success_url' => $stripe_ipn_url
89     91     'success_url' => $stripe_ipn_url,
90     92     'metadata' => array(),
91     93
92     );
93

```

```

wordpress-simple-paypal-shopping-cart/trunk/includes/wpsc-misc-functions.php
Tabular | Unified
r3339959r3440173
22     22     exit;
23     23     }
24     24     else if(isset($_REQUEST["simple_cart_stripe_ipn"]))
25     25     {
26     26     include_once( WP_CART_PATH . 'stripe.php');
27     27     wpc_handle_stripe_ipn();
28     28     exit;
29     29     }
30     30     }
31     31     if (is_admin()) {
...     ...
42     42
43     43     }
44     44     }
39
40     function wpsc_handle_wp_tasks() {
41     41     if(isset($_REQUEST["simple_cart_stripe_ipn"]))
42     42     {
43     43     include_once( WP_CART_PATH . 'stripe.php');
44     44     wpc_handle_stripe_ipn();
45     45     exit;
46     46     }
44     47     }
45     48

```

```

wordpress-simple-paypal-shopping-cart/trunk/readme.txt
Tabular | Unified
r3417800r3440173
6     6     Requires PHP: 7.4
7     7     Tested up to: 6.9
8     8     Stable tag: 5.2.4
8     8     Stable tag: 5.2.5
9     9     License: GPLv2 or later
10    10
...    ...
241   241   == Changelog ==
242   242
243   243   = 5.2.5 =
244   244   - Improved the Stripe post-checkout handling code to prevent a potential issue with some server
245   245   configurations.
- Added output escaping to the 'wpsc_display_product' shortcode parameters.

```

r3417800r3440173

246

243

247

= 5.2.4 =

244

248

- Added a new feature to auto send invoices to customers for Stripe checkout.

wordpress-simple-paypal-shopping-cart/trunk/stripe.php

Tabular | Unified

r3374067r3440173

16

16

17

17

private \$cart_id = 0;

18

19

private \$csid = '';

18

20

19

21

function __construct() {

...

...

}

304

306

305

307

306

307

308

function validate_ipn() {

308

309

309

310

```
// $this->order_id = isset($_GET["ref_id"])?$_GET["ref_id"]:0; // TODO: old code. need to
remove
```

310

311

```
$this->cart_id = isset($_GET["ref_id"])?$_GET["ref_id"]:0;
```

311

312

```
$this->cart_id = isset($_GET["ref_id"]) ? sanitize_text_field($_GET["ref_id"]): 0;
```

311

313

```
$this->csid = isset($_GET["csid"]) ? sanitize_text_field($_GET["csid"]): 0;
```

312

314

//IPN validation check

...

...

325

327

```
\Stripe\Stripe::setApiKey( $this->secret_key );
```

326

328

327

328

```
$events = \Stripe\Event::all(
```

328

329

```
array(
```

329

330

```
'type' => 'checkout.session.completed',
```

330

331

```
'created' => array(
```

331

332

```
'gte' => time() - 60 * 60,
```

332

333

```
),
```

333

334

```
)
```

334

335

```
);
```

335

336

```
$sess = false;
```

336

337

337

338

```
foreach ( $events->autoPagingIterator() as $event ) {
```

338

339

```
$session = $event->data->object;
```

339

340

```
if ( isset( $session->client_reference_id ) && $session->client_reference_id ===
```

340

341

```
$this->cart_id ) {
```

341

342

```
$sess = $session;
```

342

343

```
break;
```

343

344

```
}
```

344

329

```
$sess = $this->retrieve_checkout_session_object();
```

345

330

346

331

```
if ( false === $sess ) {
```

...

...

379

364

```
$this->debug_log( 'IPN successfully verified.', true );
```

380

365

```
return true;
```

366

```
}
```

367

368

```
/**
```

369

```
* Retrieves the stripe checkout session.
```

370

```
*
```

371

```
* It might fail in the first attempt if there is any race condition.
```

372

```
* So to prevent that, this method contains a recursion mechanism with max_attempt count.
```

r3374067r3440173		
373		*/
374		private function retrieve_checkout_session_object() {
375		// TODO: old code. need to remove
376		//\$events = \Stripe\Event::all(
377		// array(
378		// 'type' => 'checkout.session.completed',
379		// 'created' => array(
380		// 'gte' => time() - 60 * 60,
381		//),
382		//)
383		//));
384		//
385		//foreach (\$events->autoPagingIterator() as \$event) {
386		// \$session = \$event->data->object;
387		// if (isset(\$session->client_reference_id) && \$session->client_reference_id ===
388		\$this->cart_id) {
389		// \$sess = \$session;
390		// break;
391		// }
392		//}
393		\$checkout_session_id = \$this->csid;
394		\$sess = \Stripe\Checkout\Session::retrieve(\$checkout_session_id);
395		
396		if (!empty(\$sess)){
397		return \$sess;
398		}
399		
400		wpsc_log_payment_debug('The checkout session could not be retrieved. Retrying to retrieve
401		checkout session...', false);
402		sleep(2);
403		\$sess = \Stripe\Checkout\Session::retrieve(\$checkout_session_id);
404		
405		return \$sess;
381	406	}
382	407	
...	...	
604	629	}
605	630	
631		public function get_cart_id() {
632		return \$this->cart_id;
633		}
606	634	}
607	635	
...	...	
643	671	}
644	672	
645		\$cart_id = isset(\$_GET["ref_id"])?\$_GET["ref_id"]:'';
673		\$cart_id = \$ipn_handler_instance->get_cart_id();
646	674	\$redirect_url = add_query_arg('cart_id', \$cart_id, \$return_url);
647	675	\$redirect_url = add_query_arg('_wpnonce', wp_create_nonce('wpsc_thank_you_nonce_action'),
		\$redirect_url);

wordpress-simple-paypal-shopping-cart/trunk/wp_shopping_cart.php

Tabular | Unified

r3417800r3440173

3	3	/*
4	4	Plugin Name: Simple Shopping Cart
5		Version: 5.2.4
5		Version: 5.2.5

r3417800r3440173		
6	6	Plugin URI: https://www.tipsandtricks-hq.com/wordpress-simple-paypal-shopping-cart-plugin-768
7	7	Author: Tips and Tricks HQ, Ruhul Amin, mra13
...	...	
18	18	}
19	19	
20		define('WP_CART_VERSION', '5.2.4');
	20	define('WP_CART_VERSION', '5.2.5');
21	21	define('WP_CART_FOLDER', dirname(plugin_basename(__FILE__)));
22	22	define('WP_CART_PATH', plugin_dir_path(__FILE__));
...	...	
851	851	add_action('admin_enqueue_scripts', 'wpsc_admin_side_enqueue_scripts');
852	852	add_action('admin_print_styles', 'wpsc_admin_side_styles');
	853	
	854	add_action('wp', 'wpsc_handle_wp_tasks');

wordpress-simple-paypal-shopping-cart/trunk/wp_shopping_cart_shortcodes.php		Tabular	Unified
r3416198r3440173			
115	115	}	
116	116		
	117	\$description = sanitize_text_field(\$description);	
	118		
117	119	\$price = wpsc_strip_char_from_price_amount(\$price);	
118	120	\$shipping = wpsc_strip_char_from_price_amount(\$shipping);	
...	...		
138	140	<div class="wp_cart_product_display_box">	
139	141	<div class="wp_cart_product_thumbnail">	
140		<?php echo \$thumbnail_code; ?>	
	142	<?php echo wp_kses_post(\$thumbnail_code); ?>	
141	143	</div>	
142	144	<div class="wp_cart_product_display_bottom">	
143	145	<div class="wp_cart_product_name">	
144		<?php echo \$name ?>	
	146	<?php echo esc_attr(\$name) ?>	
145	147	</div>	
146	148	<div class="wp_cart_product_description">	
147		<?php echo \$description ?>	
	149	<?php echo wp_kses_post(\$description) ?>	
148	150	</div>	
149	151	<div class="wp_cart_product_price">	
150		<?php echo \$formatted_price ?>	
	152	<?php echo wp_kses_post(\$formatted_price) ?>	
151	153	</div>	
152	154	<div class="wp_cart_product_button">	

[About](#)[Showcase](#)[Learn](#)[Get Involved](#)[News](#)[Themes](#)[Documentation](#)[Events](#)[Hosting](#)[Plugins](#)[Developers](#)[Donate ↗](#)[Privacy](#)[Patterns](#)[WordPress.tv ↗](#)[Five for the Future](#)

[WordPress.com ↗](#)

[Matt ↗](#)

[bbPress ↗](#)

[BuddyPress ↗](#)



CODE IS POETRY

The WordPress® trademark is the intellectual property of the WordPress Foundation.