



Plugin Directory

source: [zip-code-based-content-protection / trunk / public / class-zipcode-bcp-public.php](#)

Last change on this file was 3464999, checked in by PressTigers, 5 weeks ago

Released security update version 1.0.3

File size: 10.0 KB

Line	
1	<code><?php</code>
2	<code>/**</code>
3	<code> * The public-facing functionality of the plugin.</code>
4	<code> *</code>
5	<code> * @link https://www.presstigers.com/</code>
6	<code> * @since 1.0.0</code>
7	<code> *</code>
8	<code> * @package Zipcode_BCP</code>
9	<code> * @subpackage Zipcode_BCP/public</code>
10	<code> */</code>
11	
12	<code>/**</code>
13	<code> * The public-facing functionality of the plugin.</code>
14	<code> *</code>
15	<code> * Defines the plugin name, version, and two examples hooks for how to</code>
16	<code> * enqueue the public-facing stylesheet and JavaScript.</code>
17	<code> *</code>
18	<code> * @package Zipcode_BCP</code>
19	<code> * @subpackage Zipcode_BCP/public</code>
20	<code> * @author PressTigers <support@presstigers.com></code>
21	<code> */</code>
22	<code>class Zipcode_BCP_Public {</code>
23	
24	<code> /**</code>
25	<code> * The ID of this plugin.</code>
26	<code> *</code>
27	<code> * @since 1.0.0</code>
28	<code> * @access private</code>
29	<code> * @var string \$plugin_name The ID of this plugin.</code>
30	<code> */</code>
31	<code> private \$plugin_name;</code>
32	
33	<code> /**</code>
34	<code> * The version of this plugin.</code>
35	<code> *</code>
36	<code> * @since 1.0.0</code>
37	<code> * @access private</code>
38	<code> * @var string \$version The current version of this plugin.</code>
39	<code> */</code>
40	<code> private \$version;</code>
41	
42	<code> /**</code>
43	<code> * Initialize the class and set its properties.</code>
44	<code> *</code>
45	<code> * @since 1.0.0</code>
46	<code> * @param string \$plugin_name The name of the plugin.</code>

```

Line
47      * @param    string $version    The version of this plugin.
48      */
49      public function __construct( $plugin_name, $version ) {
50
51          $this->plugin_name = $plugin_name;
52          $this->version     = $version;
53      }
54
55      /**
56       * Register the stylesheets for the public-facing side of the site.
57       *
58       * @since    1.0.0
59       */
60      public function zbcplib_enqueue_styles() {
61
62          wp_enqueue_style( $this->plugin_name, plugin_dir_url( __FILE__ ) . 'css/zipcode-bcp-
public.css', array(), $this->version, 'all' );
63      }
64
65      /**
66       * Register the JavaScript for the public-facing side of the site.
67       *
68       * @since    1.0.0
69       */
70      public function zbcplib_enqueue_scripts() {
71
72          wp_enqueue_script( $this->plugin_name, plugin_dir_url( __FILE__ ) . 'js/zipcode-bcp-
public.js', array( 'jquery' ), $this->version, false );
73          $options                = get_option( 'cdzc_settings_no_zipcode_added' );
74          $cdzc_settings_no_valid_email = get_option( 'cdzc_settings_no_valid_email' );
75          wp_localize_script(
76              $this->plugin_name,
77              'frontend_ajax_object',
78              array(
79                  'ajaxurl'                => admin_url( 'admin-ajax.php' ),
80                  'cdzc_settings_no_zipcode_added' => __( $options, 'zipcode-bcp' ),
81                  'cdzc_settings_no_valid_email'
=> __( $cdzc_settings_no_valid_email, 'zipcode-bcp' ),
82              )
83          );
84      }
85
86      /**
87       * Filtering the content
88       *
89       * @since    1.0.0
90       */
91      public function zbcplib_filter_page_and_post_content() {
92
93          if ( is_singular() ) {
94              global $post;
95              $post_id                = $post->ID;
96
97              $activate_zipcode_checkbox = get_post_meta( $post_id, 'activate_zipcode_checkbox', true );
98              $screens                    = get_option( 'cdzc_settings_for_checkbox' );
99              $post_type                  = get_post_type( $post_id );
100
101              if ( in_array( $post_type, $screens ) && $activate_zipcode_checkbox == 1 && ! isset( $_COOKIE[ 'zipcode-
view-content-' . $post_id ] ) ) {
102                  $url = site_url() . '/zipcode-bcp/redirect/zipcode/?id=' . $post_id . '';
103                  wp_safe_redirect( $url, '302' );
104                  exit();
105              }
106          }
107      }

```

```

Line
104         }
105     }
106
107     /**
108      * Function for setting the rule.
109      *
110      * @since 1.0.0
111      */
112     public function zbcplib_rewrite_rule() {
113         add_rewrite_rule( '^zipcode-bcp/redirect/zipcode/?$', 'index.php?pg=zipcode-
114 bcp', 'top' );
115         global $wp_rewrite;
116         $wp_rewrite->flush_rules();
117     }
118
119     /**
120      * Function for adding the query vars for the plugin.
121      *
122      * @since 1.0.0
123      */
124     public function zbcplib_register_query_var( $vars ) {
125         $vars[] = 'pg';
126         $vars[] = 'id';
127         return $vars;
128     }
129
130     /**
131      * Function for loading the template.
132      *
133      * @since 1.0.0
134      */
135     public function zbcplib_load_templates() {
136         if ( get_query_var( 'pg' ) ) {
137             if ( get_query_var( 'pg' ) == 'zipcode-bcp' ) {
138                 add_filter(
139                     'template_include',
140                     function () {
141                         return plugin_dir_path( __DIR__ ) . 'public/partials/zipcode-bcp-public-display.php';
142                     }
143                 );
144                 return;
145             }
146         }
147     }
148
149     /**
150      * function for handling the ajax request of zipcode search form
151      *
152      * @since 1.0.0
153      */
154     public function is_valid_zip_code( $zip_codes ) {
155         return ( preg_match( '/^[a-zA-Z0-9]{4,8}(-[0-9]{4})?
156 $/', $zip_codes ) ) ? true : false;
157     }
158     public function check_zipcode_from_post_page_meta() {
159
160         if (
161             ! isset( $_POST['nonce'] ) ||
162             ! wp_verify_nonce( $_POST['nonce'], 'zbcplib_check_zipcode' )
163         ) {

```

```

Line
163         wp_send_json_error( 'Invalid security token' );
164     }
165
166     $post_id           = sanitize_text_field( $_POST['post_id'] );
167     $zipcode           = sanitize_text_field( $_POST['zipcode'] );
168     $post_page_zipcode_select
= get_post_meta( $post_id, 'post_page_zipcode_select', true );
169
170 $cdzc_settings_not_providing_services = get_option( 'cdzc_settings_not_providing_services' );
171     $cdzc_settings_no_zipcode_added     = get_option( 'cdzc_settings_no_zipcode_added' );
172     $is_valid_format                     = $this-
>is_valid_zip_code( sanitize_text_field( $_POST['zipcode'] ) );
173     if ( $is_valid_format ) {
174         if ( isset( $_POST['zipcode'] ) ) {
175             if ( in_array( $zipcode, $post_page_zipcode_select ) ) {
176                 $cookie_name = 'zipcode-view-content-' . $post_id;
177                 $cookie_value = $zipcode . '-' . $post_id;
178                 setcookie( $cookie_name, $cookie_value, time() + ( 86400 * 30 ), '/' ); // 86400 = 1 day
179                 $url = get_the_permalink( $post_id );
180                 echo json_encode(
181                     array(
182                         'status'      => true,
183                         'showSearch' => false,
184                         'result'     => $url,
185                     )
186                 );
187             } else {
188                 global $wpdb;
189                 $prefix = $wpdb->prefix;
190                 // Check someone already requested for this ZIP Code
191                 $table = $prefix . 'zipcode_requested';
192                 $count = $wpdb->get_var(
193                     $wpdb->prepare(
194                         "SELECT COUNT(*) FROM {$table} WHERE zipcode =
195                         %s",
196                         $zipcode
197                     )
198                 );
199                 // Already requested by someone else
200                 if ( $this-
>zbcpc_is_cookies_exist( 'is_requested', $zipcode ) != 2 ) {
201                     // User not requested for this Zip code
202                     $this->zbcpc_set_ip_cookies( 'is_requested', $zipcode );
203                     if ( $count ) {
204                         $zip_codes = $wpdb->get_results(
205                             $wpdb->prepare(
206                                 "SELECT * FROM {$table} WHERE
207                                 zipcode = %s",
208                                 $zipcode
209                             ),
210                             ARRAY_A
211                         );
212                         $zip_codes      = $zip_codes[0];
213                         $request_count = $zip_codes['request_count'];
214                         ++$request_count;
215                         $zip_codes['request_count'] = $request_count;
216                         $wpdb->update( $wpdb-
>prefix . 'zipcode_requested', $zip_codes, array( 'id' => $zip_codes['id'] ) );
217                     } else {

```

```

Line
218         $zipcode_data = array(
219             'zipcode'      => $zipcode,
220             'request_count' => 1,
221         );
222
223         $wpdb->insert(
224             $wpdb->prefix . 'zipcode_requested',
225             $zipcode_data,
226             array( '%s', '%d' )
227         );
228     }
229 }
230 echo json_encode(
231     array(
232         'status'      => false,
233         'showSearch' => true,
234         'result'      => __(
235             $cdzc_settings_not_providing_services,
236             'zipcode-bcp'
237         ),
238     )
239 );
240 }
241 } else {
242     echo json_encode(
243         array(
244             'status'      => false,
245             'showSearch' => false,
246             'result'      => __(
247                 $cdzc_settings_no_zipcode_added,
248                 'zipcode-bcp'
249             ),
250         )
251     );
252 }
253 } else {
254     echo json_encode(
255         array(
256             'status' => false,
257             'result' => __(
258                 'Invalid Format',
259                 'zipcode-bcp'
260             ),
261         )
262     );
263 }
264 die();
265 }
266
267 // set cookies with IP
268 public function zbcg_set_ip_cookies( $ip, $zip ) {
269     $cookie_expire_on = time() + 60 * 60 * 24 * 30;
270     $cookies_exist    = $this->zbcg_is_cookies_exist( $ip, $zip );
271
272     if ( $cookies_exist == 0 ) {
273         setcookie( $ip, $zip, $cookie_expire_on, '/' );
274     } elseif ( $cookies_exist == 1 ) {
275         $zips = sanitize_text_field( $_COOKIE[ $ip ] );
276         $zips = $zips . ',' . $zip;
277         setcookie( $ip, $zips, $cookie_expire_on, '/' );
278     }

```

```

Line
279     }
280
281     /**
282     * Check cookies already exist or not
283     * 0 - Not exist
284     * 1 - Exist but not added Ip
285     * 2 - Exist with IP
286     *
287     * @since 1.0.0
288     */
289     public function zbcps_is_cookies_exist( $ip, $zip ) {
290
291         if ( ! isset( $_COOKIE[ $ip ] ) ) {
292             return 0;
293         } else {
294             $ips      = array_map( 'sanitize_text_field', $_COOKIE[ $ip ] );
295             $zipcodes = explode( ',', $ips );
296             foreach ( $zipcodes as $zipcode ) {
297                 if ( $zipcode == $zip ) {
298                     return 2;
299                 }
300             }
301             return 1;
302         }
303     }
304
305     /**
306     * Function for validation of zipcode
307     *
308     * @since 1.0.0
309     */
310
311
312     /**
313     * function for handling the ajax request of submit email form
314     *
315     * @since 1.0.0
316     */
317     public function submit_email_against_zipcode() {
318
319         if (
320             ! isset( $_POST['nonce'] ) ||
321             ! wp_verify_nonce( $_POST['nonce'], 'zbcps_submit_email_zipcode' )
322         ) {
323             wp_send_json_error( 'Invalid security token' );
324         }
325
326         $email      = sanitize_email( $_POST['email'] );
327         $zipcode     = sanitize_text_field( $_POST['zipcode'] );
328         $post_id    = intval( $_POST['post_id'] );
329         $post_type  = sanitize_text_field( $_POST['post_type'] );
330         $post_title = sanitize_title( $_POST['post_title'] );
331
332         $cdzcs_settings_not_providing_services = get_option( 'cdzcs_settings_not_providing_services' );
333         $cdzcs_settings_no_zipcode_added      = get_option( 'cdzcs_settings_no_zipcode_added' );
334
335         $cdzcs_settings_user_already_requested = get_option( 'cdzcs_settings_user_already_requested' );
336         $cdzcs_settings_thankyou_for_email    = get_option( 'cdzcs_settings_thankyou_for_email' );
337
338         global $wpdb;

```

```
Line
339     $prefix = $wpdb->prefix;
340     $table  = $prefix . 'zipcode_requested_users';
341
342     $count = $wpdb->get_var(
343         $wpdb->prepare(
344             "SELECT COUNT(*) FROM {$table} WHERE user_email = %s AND zipcode = %s",
345             $email,
346             $zipcode
347         )
348     );
349
350     if ( $count ) {
351         echo json_encode(
352             array(
353                 'status' => false,
354                 'result' => __(
355                     $cdzc_settings_user_already_requested,
356                     'zipcode-bcp'
357                 ),
358             )
359         );
360         die();
361     } else {
362         $zipcode_data = array();
363         $zipcode_data['zipcode'] = $zipcode;
364         $zipcode_data['user_email'] = $email;
365         $zipcode_data['post_id'] = $post_id;
366         $zipcode_data['post_type'] = $post_type;
367         $zipcode_data['post_title'] = $post_title;
368
369         $result = $wpdb->insert( $wpdb-
370 >prefix . 'zipcode_requested_users', $zipcode_data );
371         echo json_encode(
372             array(
373                 'status' => true,
374                 'result' => __(
375                     $cdzc_settings_thankyou_for_email,
376                     'zipcode-bcp'
377                 ),
378             )
379         );
380         die();
381     }
382     die();
383 }
```

[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.