



# Plugin Directory

## Changeset 3470119 for download-monitor

**Timestamp:** 02/26/2026 10:04:54 AM (5 weeks ago)

**Author:** altesin

**Message:** Update to version 5.1.8 from GitHub

**Location:** [download-monitor](#)

**Files:** ■ 2 added ■ 14 edited ■ 1 copied

- tags/5.1.8 (copied from *download-monitor/trunk*)
- tags/5.1.8/download-monitor.php (2 diffs)
- tags/5.1.8/readme.txt (2 diffs)
- tags/5.1.8/src/Dependencies/Pimple/Container.php (2 diffs)
- tags/5.1.8/src/Dependencies/Pimple/Tests/PimpleTest.php (5 diffs)
- tags/5.1.8/src/Shop/Checkout/PaymentGateway/PayPal/CaptureOrder.php (3 diffs)
- tags/5.1.8/src/Shop/Checkout/PaymentGateway/PayPal/ExecutePaymentListener.php (6 diffs)
- tags/5.1.8/vendor/bin
- tags/5.1.8/vendor/composer/installed.php (2 diffs)
- trunk/download-monitor.php (2 diffs)
- trunk/readme.txt (2 diffs)
- trunk/src/Dependencies/Pimple/Container.php (2 diffs)
- trunk/src/Dependencies/Pimple/Tests/PimpleTest.php (5 diffs)
- trunk/src/Shop/Checkout/PaymentGateway/PayPal/CaptureOrder.php (3 diffs)
- trunk/src/Shop/Checkout/PaymentGateway/PayPal/ExecutePaymentListener.php (6 diffs)
- trunk/vendor/bin
- trunk/vendor/composer/installed.php (2 diffs)

Unmodified  Added  Removed

download-monitor/tags/5.1.8/download-monitor.php		Tabular   Unified
r3447860r3470119		
4	4	Plugin URI: <a href="https://www.download-monitor.com">https://www.download-monitor.com</a>
5	5	Description: A full solution for managing and selling downloadable files, monitoring downloads and outputting download links and file information on your WordPress powered site.
6	6	Version: 5.1.7
6	6	Version: 5.1.8
7	7	Author: WPChill
8	8	Author URI: <a href="https://wpchill.com">https://wpchill.com</a>
...	...	
35	35	
36	36	// Define DLM Version
37	37	define('DLM_VERSION', '5.1.7');
37	37	define('DLM_VERSION', '5.1.8');
38	38	define('DLM_UPGRADER_VERSION', '4.6.0');
39	39	

download-monitor/tags/5.1.8/readme.txt		Tabular	Unified
r3447874r3470119			
4	4	Requires at least: 5.5	
5	5	Tested up to: 6.9	
6		Stable tag: 5.1.7	
	6	Stable tag: 5.1.8	
7	7	License: GPLv3	
8	8	Requires PHP: 7.4	
...	...		
115	115		
116	116	== Changelog ==	
	117	= 5.1.8 - 26.02.2026 =	
	118	Fixed: Security - PayPal payment gateway IDOR and financial logic vulnerability. Validates order hash (timing-safe), binds PayPal token to the order, and only completes the order when the transaction matches. Handles capture failures safely.	
	119		
117	120	= 5.1.7 - 27.01.2026 =	
118	121	Update: Removed unused images.	

download-monitor/tags/5.1.8/src/Dependencies/Pimple/Container.php		Tabular	Unified
r3447860r3470119			
197	197	}	
198	198		
199		\$this->protected->attach(\$callable);	
	199	\$this->protected->offsetSet(\$callable);	
200	200		
201	201	return \$callable;	
...	...		
269	269		
270	270	if (isset(\$this->factories[\$factory])) {	
271		\$this->factories->detach(\$factory);	
272		\$this->factories->attach(\$extended);	
	271	\$this->factories->offsetUnset(\$factory);	
	272	\$this->factories->offsetSet(\$extended);	
273	273	}	
274	274		

download-monitor/tags/5.1.8/src/Dependencies/Pimple/Tests/PimpleTest.php		Tabular	Unified
r3447860r3470119			
30	30	use PHPUnit\Framework\TestCase;	
31	31	use WPChill\DownloadMonitor\Dependencies\Pimple\Container;	
	32	use WPChill\DownloadMonitor\Dependencies\Pimple\ServiceProviderInterface;	
32	33		
33	34	/**	
...	...		
205	206	{	
206	207	\$pimple = new Container();	
207		\$this->assertSame(\$pimple, \$pimple->register(\$this->getMockBuilder('WPChill\DownloadMonitor\Dependencies\Pimple\ServiceProviderInterface')->getMock()));	
	208		
	209	\$stub = new class implements ServiceProviderInterface {	
	210	public function register(Container \$pimple)	
	211	{	
	212	}	
	213	};	
	214		
	215	\$this->assertSame(\$pimple, \$pimple->register(\$stub));	
208	216	}	

r3447860r3470119	
209	217
...	...
276	284
277	285
278	286
279	287
280	288
...	...
282	290
283	291
284	292
285	293
286	294
...	...
426	434
427	435
	436
428	437
429	438

```

    $p = new \ReflectionProperty($pimple, 'values');
    if (PHP_VERSION < 80100) {
    if (PHP_VERSION_ID < 80100) {
        $p->setAccessible(true);
    }

    $p = new \ReflectionProperty($pimple, 'factories');
    if (PHP_VERSION < 80100) {
    if (PHP_VERSION_ID < 80100) {
        $p->setAccessible(true);
    }

    * @group legacy
    * @expectedDeprecation How WPChill\DownloadMonitor\Dependencies\Pimple behaves when extending
    protected closures will be fixed in WPChill\DownloadMonitor\Dependencies\Pimple 4. Are you sure
    "foo" should be protected?
    * @dataProvider badServiceDefinitionProvider
    */
    #[DataProvider('badServiceDefinitionProvider')]

```

download-monitor/tags/5.1.8/src/Shop/Checkout/PaymentGateway/PayPal/CaptureOrder.php		Tabular	Unified
r3198966r3470119			
27	27	}	
28	28		
	29	/**	
	30	* Whether a response was set (e.g. capture succeeded).	
	31	*	
	32	* @return bool	
	33	*/	
	34	public function has_response() {	
	35	return isset( \$this->response->result );	
	36	}	
	37		
29	38	public function getStatus() {	
30	39	return \$this->response->result->status;	
...	...		
51	60	try {	
52	61		
53	62	\$request = new OrdersCaptureRequest( \$this->order_id );	
	62	\$request = new OrdersCaptureRequest( \$this->order_id );	
54	63	\$request->body = self::buildRequestBody();	
55	64	\$response = \$this->client->execute( \$request );	
...	...		
60	69	} catch ( PayPalHttp\HttpException \$ex ) {	
61	70		
62	71	//print_r(\$ex->getMessage());	
63	72	// Capture failed (e.g. invalid token, already captured, network error).	
	72	return null;	
64	73	}	
65	74	}	

monitor/tags/5.1.8/src/Shop/Checkout/PaymentGateway/PayPal/ExecutePaymentListener.php

r3198966r3470119

```

39 39     if ( empty( $order_id ) || empty( $order_hash ) ) {
40 40         $this->execute_failed( $order_id, $order_hash );
41 41     }
41 42     }
42 43
...
54 55     }
55 56
57 57     // Verify order_hash against the retrieved order (timing-safe) to prevent IDOR.
58 58     if ( ! hash_equals( (string) $order->get_hash(), (string) $order_hash ) ) {
59 59         $this->execute_failed( $order_id, $order_hash );
60 60         return;
61 61     }
62 62
56 63     /**
57 64     * Get payment identifier (PayPal order ID / token)
58 65     */
59 66     $token = '';
...
62 69     }
63 70
71 71     if ( empty( $token ) ) {
72 72         $this->execute_failed( $order_id, $order_hash );
73 73         return;
74 74     }
75 75
76 76     // Bind token to this order: token must match one of this order's transactions (PayPal
order ID).
77 77     $transactions = $order->get_transactions();
78 78     $token_belongs_to_order = false;
79 79     foreach ( $transactions as $transaction ) {
80 80         if ( hash_equals( (string) $transaction->get_processor_transaction_id(), (string)
$token ) ) {
81 81             $token_belongs_to_order = true;
82 82             break;
83 83         }
84 84     }
85 85     if ( ! $token_belongs_to_order ) {
86 86         $this->execute_failed( $order_id, $order_hash );
87 87         return;
88 88     }
89 89
64 90     /**
65 91     * Execute the payment
91 91     * Execute the payment
66 92     */
67 93     try {
...
71 97         ->set_order_id( $token );
72 98
73 99     $response = $capture->captureOrder();
99 100     $capture_result = $capture->captureOrder();
100 100
101 101     // Handle capture failures safely (e.g. network error, invalid token).
102 102     if ( null === $capture_result || ! $capture_result->has_response() ) {
103 103         $this->execute_failed( $order->get_id(), $order->get_hash() );
104 104         return;
105 105     }
106 106

```

r3198966r3470119		
	107	\$response = \$capture_result;
74	108	
75	109	// if payment is not approved, exit;
...	...	
81	115	* Update transaction in local database
82	116	*/
83		
84		// update the order status to 'completed'
85		\$transactions = \$order->get_transactions();
	117	// Update the transaction that belongs to this token (already validated above).
	118	\$transaction_updated = false;
	119	\$transactions = \$order->get_transactions();
86	120	foreach ( \$transactions as \$transaction ) {
87		if ( \$transaction->get_processor_transaction_id() == \$response->getId() ) {
	121	if ( hash_equals( (string) \$transaction->get_processor_transaction_id(), (string) \$token ) ) {
88	122	\$transaction->set_status( Services::get()->service(
		'order_transaction_factory' )->make_status( 'success' ) );
89	123	\$transaction->set_processor_status( \$response->getStatus() );
...	...	
96	130	
97	131	\$order->set_transactions( \$transactions );
	132	\$transaction_updated = true;
98	133	break;
99	134	}
	135	}
100	136	
	137	// Only complete the order if we actually updated a matching transaction (prevents token/amount mismatch).
	138	if ( ! \$transaction_updated ) {
	139	\$this->execute_failed( \$order->get_id(), \$order->get_hash() );
	140	return;
101	141	}
102	142	}

## download-monitor/tags/5.1.8/vendor/composer/installed.php

Tabular | Unified

r3447860r3470119		
4	4	'pretty_version' => 'dev-master',
5	5	'version' => 'dev-master',
6		'reference' => '78419fc33f1ec56ec0435c82c42a820686882568',
	6	'reference' => '589dba41f5f9f23158b08854cdc37d1ab7884249',
7	7	'type' => 'library',
8	8	'install_path' => __DIR__ . '/../../',
...	...	
14	14	'pretty_version' => 'dev-master',
15	15	'version' => 'dev-master',
16		'reference' => '78419fc33f1ec56ec0435c82c42a820686882568',
	16	'reference' => '589dba41f5f9f23158b08854cdc37d1ab7884249',
17	17	'type' => 'library',
18	18	'install_path' => __DIR__ . '/../../',

## download-monitor/trunk/download-monitor.php

Tabular | Unified

r3447860r3470119		
4	4	Plugin URI: <a href="https://www.download-monitor.com">https://www.download-monitor.com</a>
5	5	Description: A full solution for managing and selling downloadable files, monitoring downloads and outputting download links and file information on your WordPress powered site.
6		Version: 5.1.7
	6	Version: 5.1.8
7	7	Author: WPChill

r3447860r3470119		
8	8	Author URI: https://wpchill.com
...	...	
35	35	
36	36	// Define DLM Version
37		define('DLM_VERSION', '5.1.7');
	37	define('DLM_VERSION', '5.1.8');
38	38	define('DLM_UPGRADER_VERSION', '4.6.0');
39	39	

download-monitor/trunk/readme.txt		Tabular	Unified
r3447874r3470119			
4	4	Requires at least: 5.5	
5	5	Tested up to: 6.9	
6		Stable tag: 5.1.7	
	6	Stable tag: 5.1.8	
7	7	License: GPLv3	
8	8	Requires PHP: 7.4	
...	...		
115	115		
116	116	== Changelog ==	
	117	= 5.1.8 - 26.02.2026 =	
	118	Fixed: Security - PayPal payment gateway IDOR and financial logic vulnerability. Validates order hash (timing-safe), binds PayPal token to the order, and only completes the order when the transaction matches. Handles capture failures safely.	
	119		
117	120	= 5.1.7 - 27.01.2026 =	
118	121	Update: Removed unused images.	

download-monitor/trunk/src/Dependencies/Pimple/Container.php		Tabular	Unified
r3447860r3470119			
197	197	}	
198	198		
199		\$this->protected->attach(\$callable);	
	199	\$this->protected->offsetSet(\$callable);	
200	200		
201	201	return \$callable;	
...	...		
269	269		
270	270	if (isset(\$this->factories[\$factory])) {	
271		\$this->factories->detach(\$factory);	
272		\$this->factories->attach(\$extended);	
	271	\$this->factories->offsetUnset(\$factory);	
	272	\$this->factories->offsetSet(\$extended);	
273	273	}	
274	274		

download-monitor/trunk/src/Dependencies/Pimple/Tests/PimpleTest.php		Tabular	Unified
r3447860r3470119			
30	30	use PHPUnit\Framework\TestCase;	
31	31	use WPChill\DownloadMonitor\Dependencies\Pimple\Container;	
	32	use WPChill\DownloadMonitor\Dependencies\Pimple\ServiceProviderInterface;	
32	33		
33	34	/**	
...	...		
205	206	{	
206	207	\$pimple = new Container();	

r3447860r3470119	
207	<code>\$this-&gt;assertSame(\$pimple, \$pimple-&gt;register(\$this-&gt;getMockBuilder('WPChill\DownloadMonitor\Dependencies\Pimple\ServiceProviderInterface')-&gt;getMock()));</code>
208	
209	<code>\$stub = new class implements ServiceProviderInterface {</code>
210	<code>    public function register(Container \$pimple)</code>
211	<code>    {</code>
212	<code>    }</code>
213	<code>};</code>
214	
215	<code>\$this-&gt;assertSame(\$pimple, \$pimple-&gt;register(\$stub));</code>
208	<code>}</code>
209	
...	...
276	
277	<code>\$p = new \ReflectionProperty(\$pimple, 'values');</code>
278	<code>if (PHP_VERSION &lt; 80100) {</code>
286	<code>    if (PHP_VERSION_ID &lt; 80100) {</code>
279	<code>        \$p-&gt;setAccessible(true);</code>
280	<code>    }</code>
...	...
282	
283	<code>\$p = new \ReflectionProperty(\$pimple, 'factories');</code>
284	<code>if (PHP_VERSION &lt; 80100) {</code>
292	<code>    if (PHP_VERSION_ID &lt; 80100) {</code>
285	<code>        \$p-&gt;setAccessible(true);</code>
286	<code>    }</code>
...	...
426	<code>* @group legacy</code>
427	<code>* @expectedDeprecation How WPChill\DownloadMonitor\Dependencies\Pimple behaves when extending protected closures will be fixed in WPChill\DownloadMonitor\Dependencies\Pimple 4. Are you sure "foo" should be protected?</code>
436	<code>* @dataProvider badServiceDefinitionProvider</code>
428	<code>*/</code>
429	<code>#[DataProvider('badServiceDefinitionProvider')]</code>

download-monitor/trunk/src/Shop/Checkout/PaymentGateway/PayPal/CaptureOrder.php		Tabular	Unified
r3198966r3470119			
27	27	<code>}</code>	
28	28		
	29	<code>/**</code>	
	30	<code> * Whether a response was set (e.g. capture succeeded).</code>	
	31	<code> *</code>	
	32	<code> * @return bool</code>	
	33	<code> */</code>	
	34	<code>public function has_response() {</code>	
	35	<code>    return isset( \$this-&gt;response-&gt;result );</code>	
	36	<code>}</code>	
	37		
29	38	<code>public function getStatus() {</code>	
30	39	<code>    return \$this-&gt;response-&gt;result-&gt;status;</code>	
...	...		
51	60	<code>try {</code>	
52	61		
53		<code>    \$request = new OrdersCaptureRequest( \$this-&gt;order_id );</code>	
	62	<code>    \$request = new OrdersCaptureRequest( \$this-&gt;order_id );</code>	
54	63	<code>    \$request-&gt;body = self::buildRequestBody();</code>	
55	64	<code>    \$response = \$this-&gt;client-&gt;execute( \$request );</code>	
...	...		
60	69	<code>} catch ( PayPalHttp\HttpException \$ex ) {</code>	

r3198966r3470119

61	70
62	
63	
	71
	72
64	73
65	74

`//print_r($ex->getMessage());``// Capture failed (e.g. invalid token, already captured, network error).  
return null;``}``}`

download-monitor/trunk/src/Shop/Checkout/PaymentGateway/PayPal/ExecutePaymentListener.php

Tabular | Unified

r3198966r3470119

39	39
40	40
	41
41	42
42	43
...	...
54	55
55	56
	57
	58
	59
	60
	61
	62
56	63
57	
	64
58	65
59	66
...	...
62	69
63	70
	71
	72
	73
	74
	75
	76
	77
	78
	79
	80
	81
	82
	83
	84
	85
	86
	87
	88
	89
64	90
65	
	91
66	92
67	93
...	...

`if ( empty( $order_id ) || empty( $order_hash ) ) {  
    $this->execute_failed( $order_id, $order_hash );``return;``}``}``// Verify order_hash against the retrieved order (timing-safe) to prevent IDOR.  
if ( ! hash_equals( (string) $order->get_hash(), (string) $order_hash ) ) {  
    $this->execute_failed( $order_id, $order_hash );  
    return;  
}``/**``* Get payment identifier``* Get payment identifier (PayPal order ID / token)``*/``$token = '';``}``if ( empty( $token ) ) {  
    $this->execute_failed( $order_id, $order_hash );  
    return;  
}  
  
// Bind token to this order: token must match one of this order's transactions (PayPal  
order ID).  
$transactions = $order->get_transactions();  
$token_belongs_to_order = false;  
foreach ( $transactions as $transaction ) {  
    if ( hash_equals( (string) $transaction->get_processor_transaction_id(), (string)  
$token ) ) {  
        $token_belongs_to_order = true;  
        break;  
    }  
}  
if ( ! $token_belongs_to_order ) {  
    $this->execute_failed( $order_id, $order_hash );  
    return;  
}``/**``* Execute the payment``* Execute the payment``*/``try {`

r3198966r3470119		
71	97	->set_order_id( \$token );
72	98	
73		\$response = \$capture->captureOrder();
	99	\$capture_result = \$capture->captureOrder();
	100	
	101	// Handle capture failures safely (e.g. network error, invalid token).
	102	if ( null === \$capture_result    ! \$capture_result->has_response() ) {
	103	\$this->execute_failed( \$order->get_id(), \$order->get_hash() );
	104	return;
	105	}
	106	
	107	\$response = \$capture_result;
74	108	
75	109	// if payment is not approved, exit;
...	...	
81	115	* Update transaction in local database
82	116	*/
83		
84		// update the order status to 'completed'
85		\$transactions = \$order->get_transactions();
	117	// Update the transaction that belongs to this token (already validated above).
	118	\$transaction_updated = false;
	119	\$transactions = \$order->get_transactions();
86	120	foreach ( \$transactions as \$transaction ) {
87		if ( \$transaction->get_processor_transaction_id() == \$response->getId() ) {
	121	if ( hash_equals( (string) \$transaction->get_processor_transaction_id(), (string) \$token ) ) {
88	122	\$transaction->set_status( Services::get()->service(
		'order_transaction_factory' )->make_status( 'success' ) );
89	123	\$transaction->set_processor_status( \$response->getStatus() );
...	...	
96	130	
97	131	\$order->set_transactions( \$transactions );
	132	\$transaction_updated = true;
98	133	break;
99	134	}
	135	}
100	136	
	137	// Only complete the order if we actually updated a matching transaction (prevents
		token/amount mismatch).
	138	if ( ! \$transaction_updated ) {
	139	\$this->execute_failed( \$order->get_id(), \$order->get_hash() );
	140	return;
101	141	}
102	142	

download-monitor/trunk/vendor/composer/installed.php		Tabular	Unified
r3447860r3470119			
4	4	'pretty_version' => 'dev-master',	
5	5	'version' => 'dev-master',	
6		'reference' => '78419fc33f1ec56ec0435c82c42a820686882568',	
	6	'reference' => '589dba41f5f9f23158b08854cdc37d1ab7884249',	
7	7	'type' => 'library',	
8	8	'install_path' => __DIR__ . '/../..',	
...	...		
14	14	'pretty_version' => 'dev-master',	
15	15	'version' => 'dev-master',	
16		'reference' => '78419fc33f1ec56ec0435c82c42a820686882568',	
	16	'reference' => '589dba41f5f9f23158b08854cdc37d1ab7884249',	
17	17	'type' => 'library',	

r3447860r3470119

18

18

```
'install_path' => __DIR__ . '/../../',
```

[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.