RIPSTECH

Security Analysis Report

Post Expirator 2019-02-13_15:14

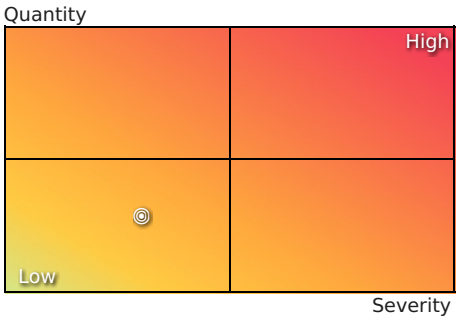
Date: 2019-02-13
User: Konstantinos Kokkorogiannis

1. Executive Summary

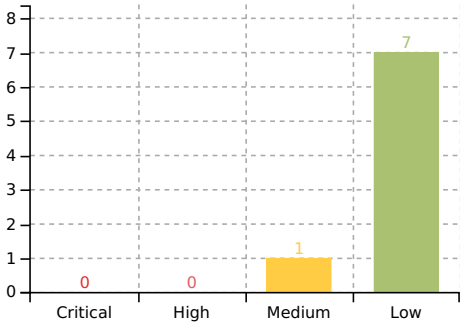
Project Name:	Post Expirator 2019-02-13_15:14
Analysis Start Date:	2019-02-13, 09:47
Analysis End Date:	2019-02-13, 09:47
Analysis Time:	18s
Engine Version:	2.13.1

Analyzed Files:	2
Analyzed LOC:	1,784
Analyzed Issue Types:	210
Detected Issues:	8
Max Issues per Type:	250

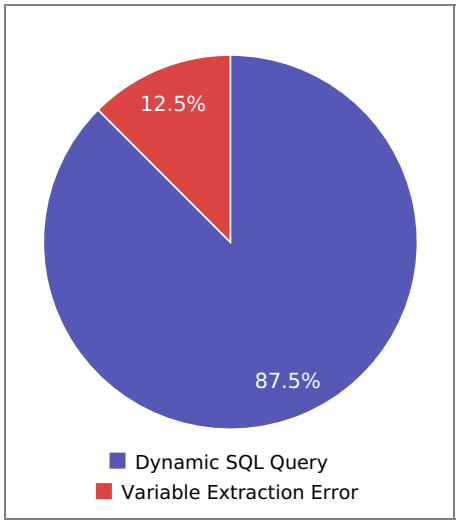
Risk Matrix



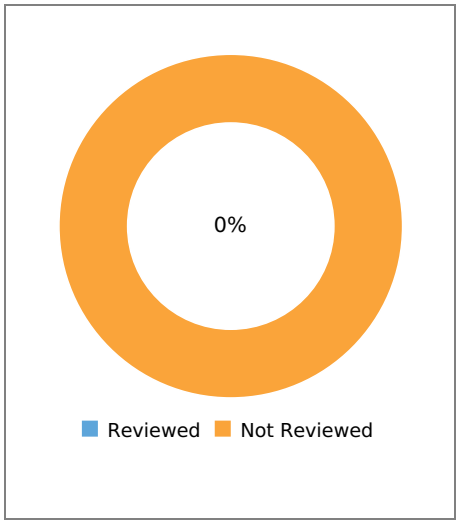
Vulnerabilities by Risk



Top Vulnerability Types



Review Status



2. Issue Breakdown

The detected security issues in this project are categorized as follows.

Severity	Vulnerability Type	CWE [?]	OWASP Top 10 [?]	SANS 25 [?]	PCI DSS [?]	ASVS [?]	Issues
Medium	Variable Extraction Error	621		Not Ranked			1
Low	Dynamic SQL Query	89		Not Ranked			7

3. Issue Details

In the following, all security issues detected in the analyzed project are presented in detail. The issues are grouped by vulnerability type and by the detected markup context. A *markup context* represents the position of user-supplied data (*source*) used in a sensitive operation (*sink*). Depending on the markup context, an attacker can alter the operation and different security mechanisms must be applied in order to patch the security issue thoroughly.

3.1. Variable Extraction Error

CWE: 621 **OWASP Top 10:** **Severity:** Medium

The application overwrites variables based on untrusted user input. This can be abused by attackers to initialize or overwrite critical internal variables and to change the control flow of the application or to exploit security-sensitive operations.

Issue #2536 - post-expirator/post-expirator.php: 625

Path: post-expirator/post-expirator.php
Line: 625
Sink: extract
Source: _POST
Taint: HTTP

Code Summary

The POST parameter '_expiration-date-options' is received in line 624 of the file post-expirator/post-expirator.php in the function postExpiratorExpire().

A code quality issue was detected in line 625 of the file post-expirator/post-expirator.php in the function postExpiratorExpire(). Please refer to the context and description for further information.

post-expirator/post-expirator.php

```
607 function postExpiratorExpire($id)
    {
⋮   ⋮
624 $postoptions = get_post_meta($id, '_expiration-date-options', true);
625 extract($postoptions);
```

Code Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
$_POST['_expiration-date-options']
```

3.2. Dynamic SQL Query

CWE: 89 **OWASP Top 10:** **Severity:** Low

A SQL query is constructed dynamically by concatenation. This can lead to SQL injection attacks. It is recommended to use prepared statements for all SQL queries. The prepared statement itself should only use placeholders for data and never concatenate data directly into the query.

Issue #2534 - post-expirator/post-expirator-debug.php: 17

Path: post-expirator/post-expirator-debug.php
Line: 17
Sink: get_var
Source:
Taint: HTTP

Code Summary

A code quality issue was detected in line 17 of the file post-expirator/post-expirator-debug.php in the method postExpiratorDebug::createDBTable(). Please refer to the context and description for further information.

post-expirator/post-expirator-debug.php

```
    class postExpiratorDebug
14  {
    private function createDBTable()
    {
15  global $wpdb;
⋮   ⋮
17  $wpdb->get_var("SHOW TABLES LIKE '" . $this->debug_table . "'");
```

SQL Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
SHOW TABLES LIKE '$RIPS_0_1_'
```

Issue #2535 - post-expirator/post-expirator-debug.php: 40

Path: post-expirator/post-expirator-debug.php
Line: 40
Sink: execute
Source:
Taint: HTTP

Code Summary

A code quality issue was detected in line 40 of the file post-expirator/post-expirator-debug.php in the method postExpiratorDebug::save(). Please refer to the context and description for further information.

post-expirator/post-expirator-debug.php

```
    class postExpiratorDebug
34  {
    public function save($data)
    {
⋮   ⋮
40  $wpdb->prepare('INSERT INTO ' . $this->debug_table . ' (`timestamp`,`message`,`blog`) VALUES (FROM_UNIXTIME(%d),%s,%s)',
    time(), $data['message'], $blog);
```

SQL Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
INSERT INTO $RIPS_0_1_ (`timestamp`,`message`,`blog`) VALUES (FROM_UNIXTIME(%d),%s,%s)
```

Issue #2537 - post-expirator/post-expirator.php: 1470

Path: post-expirator/post-expirator.php
Line: 1470
Sink: execute
Source:
Taint: HTTP

Code Summary

A code quality issue was detected in line 1470 of the file post-expirator/post-expirator.php in the function postexpirator_upgrade(). Please refer to the context and description for further information.

post-expirator/post-expirator.php

```
    function postexpirator_upgrade()
1449{
⋮   ⋮
1467global $wpdb;
⋮   ⋮
1470$wpdb->prepare('select post_id, meta_value from ' . $wpdb->postmeta . ' as postmeta, ' . $wpdb->posts . ' as posts whe
    re postmeta.post_id = posts.ID AND postmeta.meta_key = %s AND postmeta.meta_value >= %d', 'expiration-date', time());
```

SQL Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
select post_id, meta_value from $RIPS_G_7_ as postmeta, $RIPS_G_8_ as posts where postmeta.post_id = posts.ID AND
postmeta.meta_key = %s AND postmeta.meta_value >= %d
```

Issue #2538 - post-expirator/post-expirator.php: 1491

Path: post-expirator/post-expirator.php
Line: 1491
Sink: execute
Source:
Taint: HTTP

Code Summary

A code quality issue was detected in line 1491 of the file post-expirator/post-expirator.php in the function postexpirator_upgrade(). Please refer to the context and description for further information.

post-expirator/post-expirator.php

```
1449 function postexpirator_upgrade()
     {
     ⋮
1491 $wpdb->prepare("UPDATE {$wpdb->postmeta} SET meta_key = %s WHERE meta_key = %s", '_expiration-date', 'expiration-date'
     );
```

SQL Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
UPDATE $RIPS_G_7_ SET meta_key = %s WHERE meta_key = %s
```

Issue #2539 - post-expirator/post-expirator-debug.php: 62

Path: post-expirator/post-expirator-debug.php
Line: 62
Sink: execute
Source:
Taint: HTTP

Code Summary

A code quality issue was detected in line 62 of the file post-expirator/post-expirator-debug.php in the method postExpiratorDebug::purge(). Please refer to the context and description for further information.

post-expirator/post-expirator-debug.php

```
   class postExpiratorDebug
60 {
   public function purge()
   {
61 global $wpdb;
62 $wpdb->query("TRUNCATE TABLE {$this->debug_table}");
```

SQL Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
TRUNCATE TABLE $RIPS_O_1_
```

Issue #2540 - post-expirator/post-expirator-debug.php: 45

Path: post-expirator/post-expirator-debug.php
Line: 45
Sink: get_results
Source:
Taint: HTTP

Code Summary

A code quality issue was detected in line 45 of the file post-expirator/post-expirator-debug.php in the method postExpiratorDebug::getTable(). Please refer to the context and description for further information.

post-expirator/post-expirator-debug.php

```
    class postExpiratorDebug
43  {
    public function getTable()
    {
44  global $wpdb;
45  $wpdb->get_results("SELECT * FROM {$this->debug_table} ORDER BY `id` DESC");
```

SQL Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
SELECT * FROM $RIPS_0_1_ ORDER BY `id` DESC
```

Issue #2541 - post-expirator/post-expirator-debug.php: 31

Path: post-expirator/post-expirator-debug.php
Line: 31
Sink: execute
Source:
Taint: HTTP

Code Summary

A code quality issue was detected in line 31 of the file post-expirator/post-expirator-debug.php in the method postExpiratorDebug::removeDBTable(). Please refer to the context and description for further information.

post-expirator/post-expirator-debug.php

```
    class postExpiratorDebug
29  {
    public function removeDBTable()
    {
30  global $wpdb;
31  $wpdb->query('DROP TABLE IF EXISTS ' . $this->debug_table);
```

SQL Context

The following snippet(s) do not represent actual code but the tainted markup context.

```
DROP TABLE IF EXISTS $RIPS_0_1_
```