opentext™

Fortify Security Report

12/18/25

Executive Summary

Issues Overview

On 2025. 12. 18., a source code review was performed over the TEST_01 code base. 2 files, 45 LOC (Executable) were scanned and reviewed for defects that could lead to potential security vulnerabilities. A total of 6 reviewed findings were uncovered during the analysis.

Issues by Fortify Priority Order	
Critical	6

Recommendations and Conclusions

The Issues Category section provides Fortify recommendations for addressing issues at a generic level. The recommendations for specific fixes can be extrapolated from those generic recommendations by the development group.

Project Summary

Code Base Summary

Code location: /Volumes/Moon/1125/vulnscanner/results/2025/12/18/TEST_01/source/javascript_extracted/javascript
Number of Files: 2
Lines of Code: 45
Build Label: <No Build Label>

Scan Information

Scan time: 00:30
SCA Engine version: 25.3.0.0014
Machine Name: munchanghyeon-ui-MacBookAir
Username running scan: munchanghyeon

Results Certification

Results Certification Valid

Details:

Results Signature:

SCA Analysis Results has Valid signature

Rules Signature:

There were no custom rules used in this scan

Attack Surface

Attack Surface:
Private Information:
null.null.null

System Information:
null.null.resolve

Filter Set Summary

Current Enabled Filter Set:
Security Auditor View

Filter Set Details:

Folder Filters:
If [fortify priority order] contains critical Then set folder to Critical
If [fortify priority order] contains high Then set folder to High
If [fortify priority order] contains medium Then set folder to Medium
If [fortify priority order] contains low Then set folder to Low

Audit Guide Summary

Audit guide not enabled

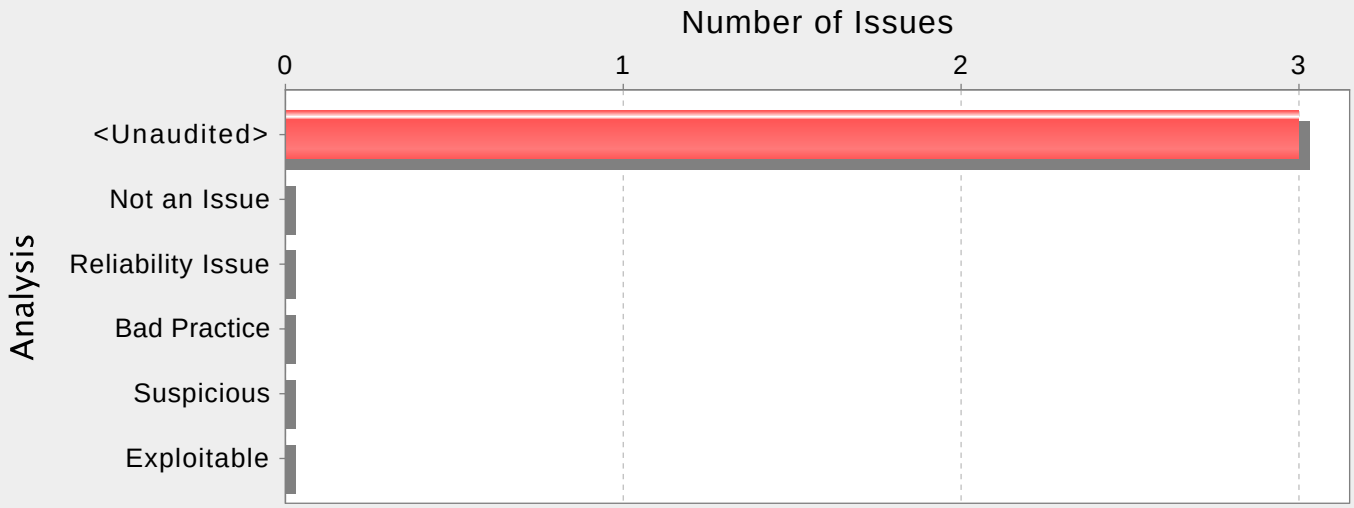
Results Outline

Overall number of results

The scan found 6 issues.

Vulnerability Examples by Category

Category: Cross-Site Scripting: DOM (3 Issues)



Abstract:

module.js changeAnchor() 6 .

Explanation:

XSS(Cross-site scripting) .

1. . DOM-based XSS , URL . Reflected XSS , Persisted(Stored) XSS .

2. . DOM-based XSS , HTML DOM(Document Object Model) .

JavaScript HTML, Flash . XSS , .

1: JavaScript URL ID eid .

```
<SCRIPT>
var pos=document.URL.indexOf("eid=")+4;
document.write(document.URL.substring(pos,document.URL.length));
</SCRIPT>
```

2: HTML .

```
<div id="myDiv">
Employee ID: <input type="text" id="eid"><br>
...
<button>Show results</button>
</div>
<div id="resultsDiv">
...
</div>
```

jQuery HTML ID .

```
$(document).ready(function(){
$("#myDiv").on("click", "button", function(){
var eid = $("#eid").val();
$("resultsDiv").append(eid);
...
```

});
});

ID eid ID . eid , HTTP .

3: React DOM-based XSS .

```
let element = JSON.parse(getUntrustedInput());
ReactDOM.render(<App>
{element}
</App>);
```

Example 3 getUntrustedInput() JSON React element Cross-Site Scripting dangerouslySetInnerHTML .

. ? URL URL . . Reflected XSS .

, XSS HTTP . XSS .

- HTTP HTTP . XSS . . URL . URL URL . , .

- . . Persistent XSS . . . .

- .

Recommendations:

XSS .

XSS . ( ) . XSS .

SQL injection . XSS . XSS . . . , XSS .

XSS HTTP . , 0-9 . . HTML .

. . . HTML HTML . XSS . SEI(Software Engineering Institute) CERT(R) Coordination Center [1].

Block-level element ( ):

- "<"
- "&" .
- ">" "<" .

.

- .
- .
- .
- "&" .

, URL . URL .

- , URL .
- "&" CGI .
- ASCII (, ISO-8859-1 127) URL .
- "%" HTTP . , "%" "%68%65%6C%6C%6F" "hello" .

<SCRIPT> </SCRIPT> :

- , , .

:

- (!) (") .

:

- UTF-7 "<" '+ADw-' . (, UTF-7) .

XSS . , . . .

, . ISO 8859-1 HTML [2].

Cross-Site Scripting HTTP Cross-Site Scripting . . . .

Tips:

1. Fortify Secure Coding Rulepacks SQL Injection , XSS . , DATABASE . .

2. URL XSS , JavaScript DOM(Document Object Model) . Rulepacks Cross-Site Scripting URL . URL Fortify Cross-Site Scripting: Poor Validation .

3. React cross-site scripting . Symbols React . Symbol (polyfills) . Cross-Site Scripting React .

module.js, line 6 (Cross-Site Scripting: DOM)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		
Abstract:	module.js changeAnchor() 6 .		

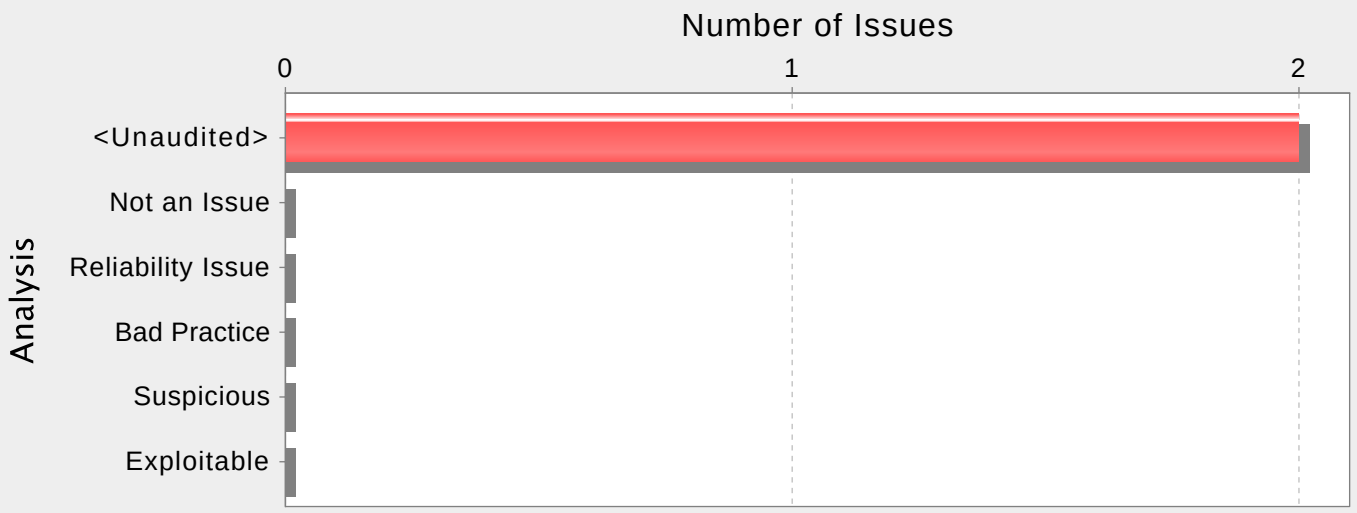
Source: module.js:6 Read document.cookie()

```
4
5   export function changeAnchor() {
6       document.write(document.cookie);
7       document.getElementById('myAnchor').href=document.cookie;
8   }
```

Sink: module.js:6 write()

```
4
5   export function changeAnchor() {
6       document.write(document.cookie);
7       document.getElementById('myAnchor').href=document.cookie;
8   }
```

Category: Privacy Violation (2 Issues)



Abstract:
module.js 6 . .

Explanation:
Privacy violation .

1. .

2. , file system .

1: .

```
localStorage.setItem('password', password);
```

. .

.

-

-

-

. , ID . ID ID .

. . .

. , . . , 2004, AOL 9 2 [1].

. , .

- (Safe Harbor Privacy Framework)[3]

- GLBA(Gramm-Leach Bliley Act)[4]

- HIPAA(Health Insurance Portability and Accountability Act)[5]

- California SB-1386 [6]

privacy violation .

Recommendations:
, . (cleanse).

. . , . . .

. , . .

Tips:
1. Privacy Violation . . privacy violation .

module.js, line 6 (Privacy Violation)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Security Features		

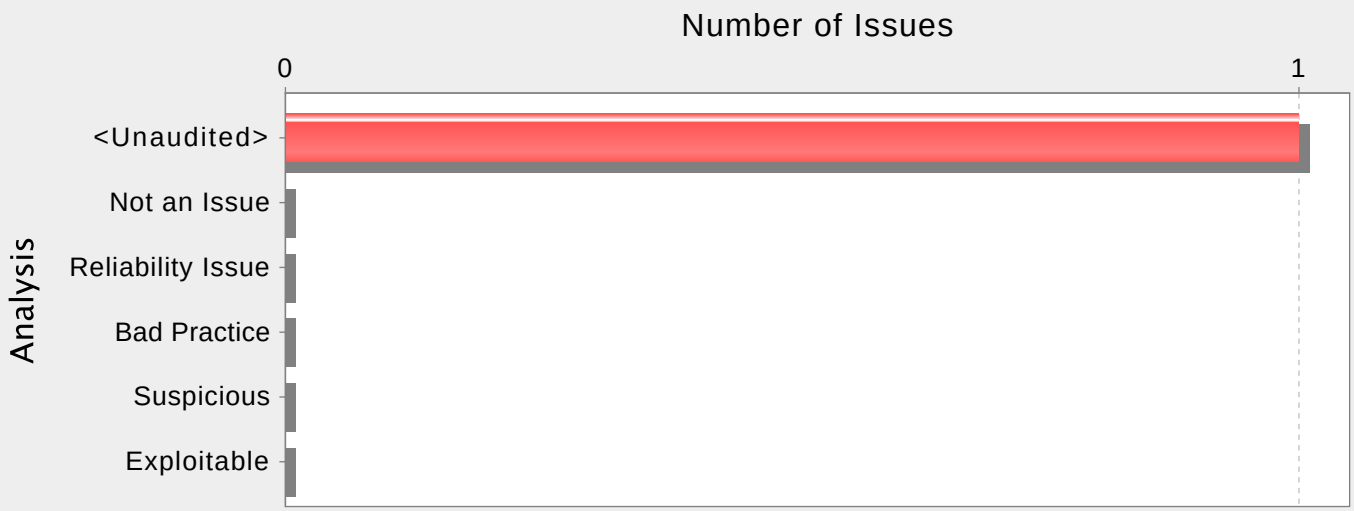
Abstract:	module.js 6 . .
Source:	module.js:6 Read document.cookie()

```
4
5                   export function changeAnchor() {
6                       document.write(document.cookie);
7                       document.getElementById('myAnchor').href=document.cookie;
8                   }
```

Sink:	module.js:6 write()

```
4
5                   export function changeAnchor() {
6                       document.write(document.cookie);
7                       document.getElementById('myAnchor').href=document.cookie;
8                   }
```

Category: Open Redirect (1 Issues)



Abstract:

module.js 7 HTTP . URL .

Explanation:

. , . URL open redirection .

Open Redirection URL . URL URL URL . open redirection .

1: JavaScript dest URL .

```
...
strDest = form.dest.value;
window.open(strDest,"myresults");
...
```

“http://trusted.example.com/ecommerce/redirect.asp?dest=www.wilyhacker.com” , . Example 1 “http://www.wilyhacker.com” .

URL . URL "http://trusted.example.com/ecommerce/redirect.asp?dest=%77%69%6C%79%68%61%63%6B%65%72%2E%63%6F%6D" .

Recommendations:

URL . . , URL . URL .

2: URL . URL .

```
...
strDest = form.dest.value;
if((strDest.value != null)||(strDest.value.length!=0))
{
if((strDest >= 0) && (strDest <= strURLArray.length -1 ))
{
strFinalURL = strURLArray[strDest];
window.open(strFinalURL,"myresults");
}
}
...
```

, URL . , .

module.js, line 7 (Open Redirect)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		

Abstract:	module.js 7 HTTP . URL .
Source:	module.js:7 Read document.cookie()

```
5  export function changeAnchor() {
6      document.write(document.cookie);
7      document.getElementById('myAnchor').href=document.cookie;
8  }
9
```

Sink:	module.js:7 Assignment to href()

```
5  export function changeAnchor() {
6      document.write(document.cookie);
7      document.getElementById('myAnchor').href=document.cookie;
8  }
9
```

Issue Count by Category

Issues by Category	
Cross-Site Scripting: DOM	3
Privacy Violation	2
Open Redirect	1

Issue Breakdown by Analysis

Issues by Analysis