Fortify Audit Workbench

Developer Workbook

SE-0019_20251127

Table of Contents

© Copyright 2008-2025 Open Text.

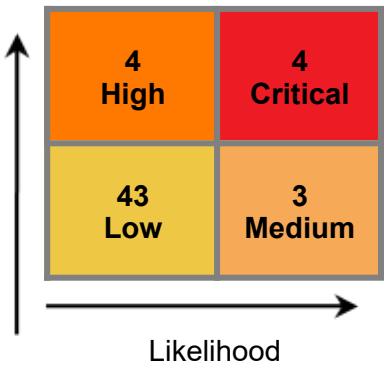
Executive Summary

This workbook is intended to provide all necessary details and information for a developer to understand and remediate the different issues discovered during the SE-0019_20251127 project audit. The information contained in this workbook is targeted at project managers and developers.

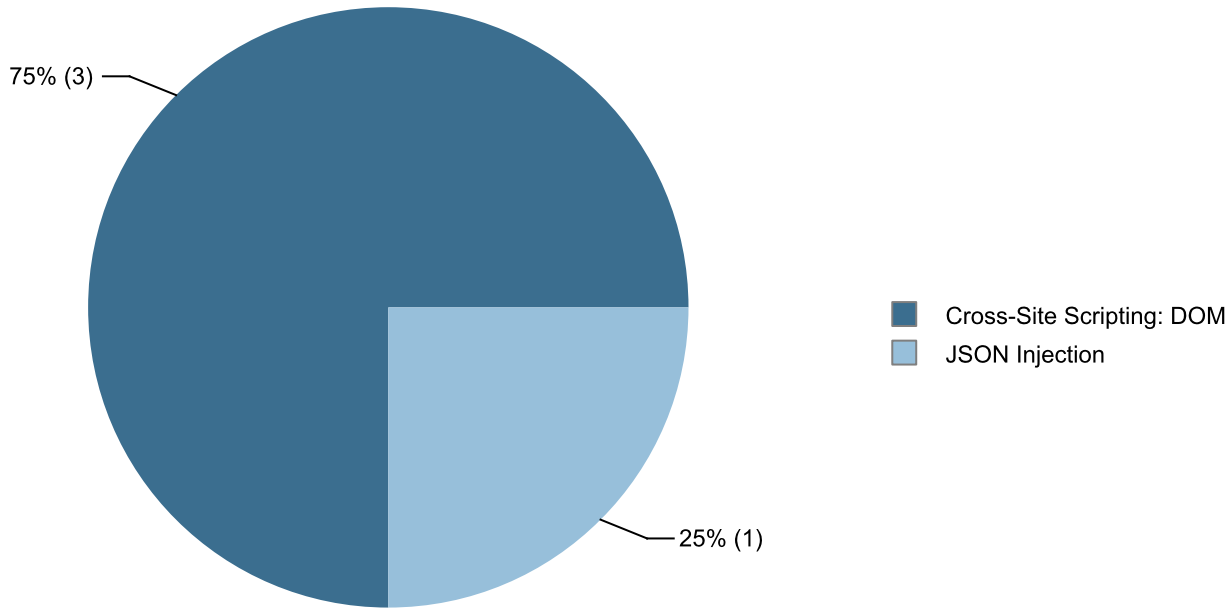
This section provides an overview of the issues uncovered during analysis.

Project Name:	SE-0019_20251127
Project Version:	
SCA:	Results Present
WebInspect:	Results Not Present
WebInspect Agent:	Results Not Present
Other:	Results Not Present

Issues by Priority



Top Ten Critical Categories



© Copyright 2008-2025 Open Text.

Project Description

This section provides an overview of the Fortify scan engines used for this project, as well as the project meta-information.

SCA

Date of Last Analysis:	2025年11月27日 上午9:32	**Engine Version:**	24.4.0.0114
Host Name:	Win2019-sca-new	**Certification:**	VALID
Number of Files:	131	**Lines of Code:**	65,783

Rulepack Name	Rulepack Version
Fortify 安全編碼規則、社群、Cloud	2025.2.1.0001
Fortify 安全編碼規則、社群、PHP	2025.2.1.0001
Fortify 安全編碼規則、社群、Universal	2025.2.1.0001
Fortify 安全編碼規則、核心、Cloud	2025.2.1.0001
Fortify 安全編碼規則、核心、JavaScript	2025.2.1.0001
Fortify 安全編碼規則、核心、PHP	2025.2.1.0001
Fortify 安全編碼規則、核心、Universal	2025.2.1.0001
Fortify 安全編碼規則、延伸、配置	2025.2.1.0001
Fortify 安全編碼規則、延伸、內容	2025.2.1.0001
Fortify 安全編碼規則、延伸、JavaScript	2025.2.1.0001

© Copyright 2008-2025 Open Text.

Issue Breakdown by Fortify Categories

The following table depicts a summary of all issues grouped vertically by Fortify Category. For each category, the total number of issues is shown by Fortify Priority Order, including information about the number of audited issues.

Category	Fortify Priority (audited/total)				Total Issues
	Critical	High	Medium	Low	
Cross-Site Request Forgery	0	0	0	0 / 28	0 / 28
Cross-Site Scripting: DOM	0 / 3	0	0	0	0 / 3
Hidden Field	0	0	0	0 / 13	0 / 13
Insecure Randomness	0	0 / 2	0	0	0 / 2
Insecure Transport: External Link	0	0	0 / 3	0	0 / 3
JSON Injection	0 / 1	0	0	0	0 / 1
Often Misused: File Upload	0	0	0	0 / 1	0 / 1
Possible Variable Overwrite: Global Scope	0	0 / 2	0	0	0 / 2
System Information Leak: Internal	0	0	0	0 / 1	0 / 1

© Copyright 2008-2025 Open Text.

Results Outline

Cross-Site Request Forgery (28 issues)

Abstract

表單發佈必須包含特定使用者機密，以避免攻擊者作出未經授權的要求。

Explanation

防禦跨網站偽造要求 (CSRF) 弱點會在以下情況中出現： 1. Web 應用程式使用了階段作業 Cookie。 2. 應用程式在回應 HTTP 要求時，沒有驗證該要求是否經使用者同意產生。 nonce 是與訊息一起傳送的加密隨機值，用於防止重複進行的攻擊。如果 HTTP 要求目前沒有包含其來源的證明，則負責處理 HTTP 要求的程式碼會容易受到 CSRF 的攻擊 (除非它不改變應用程式的狀態)。這代表使用階段作業 Cookie 的 Web 應用程式必須特別留意，以確保攻擊者沒辦法傳遞假造的要求去欺騙使用者。想像若 Web 應用程式允許管理人員提交以下表單建立新帳戶：

```
<form method="POST" action="/new_user" >
Name of new user: <input type="text" name="username">
Password for new user: <input type="password" name="user_passwd">
<input type="submit" name="action" value="Create User">
</form>
```

攻擊者可能建立如以下的網站：

```
<form method="POST" action="http://www.example.com/new_user">
<input type="hidden" name="username" value="hacker">
<input type="hidden" name="user_passwd" value="hacked">
</form>
<script>
document.usr_form.submit();
</script>
```

如果 example.com 管理員在網站上的有效階段作業期間造訪惡意頁面，則會在不知不覺中為攻擊者建立帳戶。這就是 CSRF 攻擊。這樣是有可能的，因為應用程式無法確定要求的來源。因此不論是使用者建立的或是由攻擊者建立的偽造操作，都會被視為合法的操作。攻擊者不會看到假造的要求所產生的網頁，所以這種攻擊技術只有對修改應用程式狀態的要求有用。 在 URL 中傳送階段作業識別碼而非當作 Cookie 的應用程式不會有 CSRF 問題，因為攻擊者沒有辦法存取階段作業識別碼做為假造的要求。

Recommendation

使用階段作業 Cookie 的應用程式必須包含每個發佈表單的一部分資訊，以供後端程式碼用來驗證要求的來源。其中一種方法是包含一個隨機要求識別碼或 nonce，如下所示：

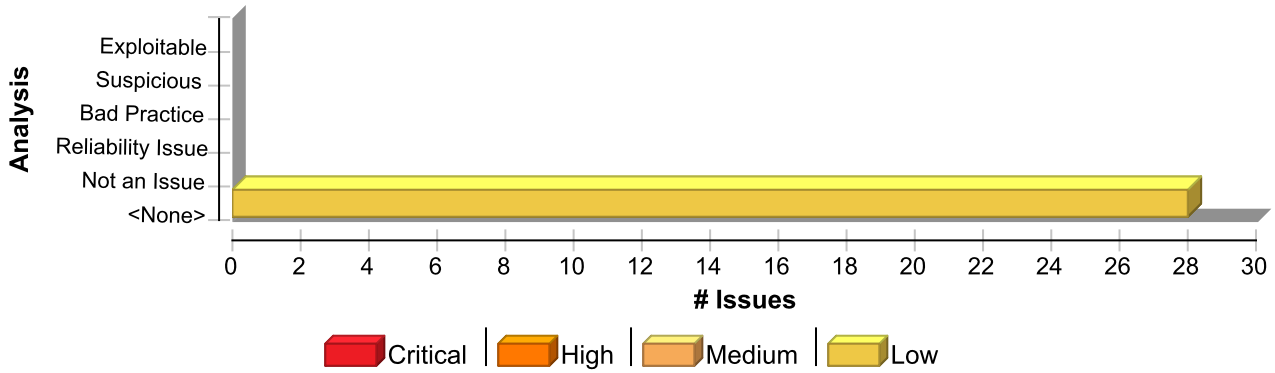
```
RequestBuilder rb = new RequestBuilder(RequestBuilder.POST, "/new_user");
body = addToPost(body, new_username);
body = addToPost(body, new_passwd);
body = addToPost(body, request_id);
rb.sendRequest(body, new NewAccountCallback(callback));
```

之後後端邏輯就能夠在處理其他的表單資料前，先驗證要求識別碼。如果可能，各伺服器要求應具備唯一的要求識別碼，而非讓特定工作階段的所有要求共用要求識別碼。就階段作業識別碼而言，攻擊者如果愈難猜出要求識別碼，就愈難發起一次成功的 CSRF 攻擊。權杖應無法輕易猜出，且其受保護的方式應與保護工作階段權杖的方式相同 (例如使用 SSLv3)。 其他減緩技術包括： **架構保護：**大多數現代 Web 應用程式架構都內嵌 CSRF 保護，且會自動包括並驗證 CSRF 權杖。 **使用挑戰/回應控制項：**強制客戶回應伺服器傳送的挑戰是針對 CSRF 的強大防禦措施。可用於此用途的一些挑戰有：CAPTCHA、密碼重新驗證和一次性權杖。 **檢查 HTTP Referer/Origin 表頭：**攻擊者將無法在執行 CSRF 攻擊時欺騙這些表頭。因此，這些表頭將成為防止 CSRF 攻擊的有用方式。 **對階段作業 Cookie 進行雙重提交：**除了傳送實際的階段作業 ID Cookie 外，還以隱藏表單值的形式傳送階段作業 ID Cookie，這是針對 CSRF 攻擊的良好保護方式。伺服器會先檢查這兩個值，確保其完全相同，然後再處理其餘的表單資料。若攻擊者代表使用者提交表單，將無法根據相同來源策略修改階段作業 ID Cookie 值。 **限制階段作業存留期：**若使用 CSRF 攻擊存取受保護的資源，只有在作

© Copyright 2008-2025 Open Text.

為攻擊的一部分而傳送的階段作業 ID 在伺服器上仍有效時，攻擊才有效。限制階段作業存留期會降低攻擊成功的可能。 XSS 攻擊會破解此處描述的技術。有效的 CSRF 減緩措施包括 XSS 減緩技術。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cross-Site Request Forgery	28	0	0	28
Total	**28**	**0**	**0**	**28**

Cross-Site Request Forgery — Low

Package: filebird.assets.js

filebird/assets/js/setting.js, line 308 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: post
Enclosing Method: lambda()
File: filebird/assets/js/setting.js:308
Taint Flags:

```
305  if (slider.length) {
306  slider.addClass("njt_loading");
307  }
308  jQuery
309  .post("options.php", data)
310  .success(function (res) {
311  if (slider.length) {
```

filebird/assets/js/setting.js, line 75 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: filebird.assets.js

filebird/assets/js/setting.js, line 75 (Cross-Site Request Forgery)

Sink Details

Sink: AssignmentStatement
Enclosing Method: insert_folder()
File: filebird/assets/js/setting.js:75
Taint Flags:

```
72  dataType: "json",
73  contentType: "application/json",
74  url: fbv_data.json_url + "/fb-insert-old-data",
75  method: "POST",
76  headers: {
77  "X-WP-Nonce": fbv_data.rest_nonce,
78  },
```

filebird/assets/js/setting.js, line 55 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: get_folders()
File: filebird/assets/js/setting.js:55
Taint Flags:

```
52  dataType: "json",
53  contentType: "application/json",
54  url: fbv_data.json_url + "/fb-get-old-data",
55  method: "POST",
56  headers: {
57  "X-WP-Nonce": fbv_data.rest_nonce,
58  },
```

filebird/assets/js/setting.js, line 111 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: lambda()
File: filebird/assets/js/setting.js:111
Taint Flags:

```
108  dataType: "json",
109  contentType: "application/json",
```


© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: filebird.assets.js

filebird/assets/js/setting.js, line 111 (Cross-Site Request Forgery)

```
110  url: fbv_data.json_url + "/fb-wipe-old-data",
111  method: "POST",
112  headers: {
113  "X-WP-Nonce": fbv_data.rest_nonce,
114  },
```

filebird/assets/js/setting.js, line 140 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: lambda()
File: filebird/assets/js/setting.js:140
Taint Flags:

```
137  dataType: "json",
138  contentType: "application/json",
139  url: fbv_data.json_url + "/fb-wipe-clear-all-data",
140  method: "POST",
141  headers: {
142  "X-WP-Nonce": fbv_data.rest_nonce,
143  },
```

filebird/assets/js/setting.js, line 189 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: lambda()
File: filebird/assets/js/setting.js:189
Taint Flags:

```
186  dataType: "json",
187  contentType: "application/json",
188  url: fbv_data.json_url + "/fb-import",
189  method: "POST",
190  cache: false,
191  headers: {
192  "X-WP-Nonce": fbv_data.rest_nonce,
```


© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: filebird.assets.js

filebird/assets/js/setting.js, line 509 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: lambda()
File: filebird/assets/js/setting.js:509
Taint Flags:

```
506  jQuery
507  .ajax({
508  url: fbv_data.json_url + "/import-csv",
509  method: "POST",
510  processData: false,
511  cache: false,
512  contentType: false,
```

filebird/assets/js/setting.js, line 589 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: lambda()
File: filebird/assets/js/setting.js:589
Taint Flags:

```
586  jQuery
587  .ajax({
588  url: fbv_data.json_url + "/generate-attachment-size",
589  method: "POST",
590  cache: false,
591  dataType: "json",
592  contentType: "application/json",
```

filebird/assets/js/setting.js, line 629 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: lambda()

© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: filebird.assets.js

filebird/assets/js/setting.js, line 629 (Cross-Site Request Forgery)

File: filebird/assets/js/setting.js:629
Taint Flags:

```
626  jQuery
627  .ajax({
628  url: window.ajaxurl,
629  method: "POST",
630  cache: false,
631  data: {
632  action: "fbv_sync_wpml",
```

filebird/assets/js/yc.js, line 5 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: lambda()
File: filebird/assets/js/yc.js:5
Taint Flags:

```
2  jQuery('#njt-yc-noti-dismiss').click(function(){
3  jQuery.ajax({
4  url: ajaxurl,
5  method: 'POST',
6  data: {
7  action: 'njt_yaycommerce_dismiss',
8  nonce: yaycommerce.nonce,
```

filebird/assets/js/setting.js, line 231 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: import_site()
File: filebird/assets/js/setting.js:231
Taint Flags:

```
228  dataType: "json",
229  contentType: "application/json",
230  url: fbv_data.json_url + "/fb-import-insert-folder",
231  method: "POST",
232  cache: false,
```


© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: filebird.assets.js

filebird/assets/js/setting.js, line 231 (Cross-Site Request Forgery)

```
233  headers: {
234  "X-WP-Nonce": fbv_data.rest_nonce,
```

filebird/assets/js/setting.js, line 250 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: import_site()
File: filebird/assets/js/setting.js:250
Taint Flags:

```
247  dataType: "json",
248  contentType: "application/json",
249  url: fbv_data.json_url + "/fb-import-after-inserting",
250  method: "POST",
251  cache: false,
252  data: JSON.stringify({
253  site: site,
```

filebird/assets/js/setting.js, line 391 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: AssignmentStatement
Enclosing Method: getUserFromCSV()
File: filebird/assets/js/setting.js:391
Taint Flags:

```
388  return jQuery
389  .ajax({
390  url: fbv_data.json_url + "/import-csv-detail",
391  method: "POST",
392  processData: false,
393  cache: false,
394  contentType: false,
```

Package: filebird.views.pages

filebird/views/pages/html-settings.php, line 63 (Cross-Site Request Forgery)

Issue Details

© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: filebird.views.pages

filebird/views/pages/html-settings.php, line 63 (Cross-Site Request Forgery)

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: filebird/views/pages/html-settings.php:63
Taint Flags:

```
60 ?>
61 <div class="wrap">
62  <h1><?php esc_html_e( 'FileBird Settings', 'filebird' ); ?></h1>
63  <form action="options.php" method="POST" id="fbv-setting-form" autocomplete="off">
64  <?php settings_fields( 'njt_fbv' ); ?>
65  <?php do_settings_sections( 'njt_fbv' ); ?>
66  <nav class="nav-tab-wrapper">
```

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 52 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:52
Taint Flags:

```
49
50  <div id="ure_tabs-1">
51  <div id="ure-settings-form">
52  <form method="post" action="<?php echo $link; ?>?page=settings-<?php echo
URE_PLUGIN_FILE; ?>" >
53  <table id="ure_settings">
54 <?php
55 if ( ! $license_key_only ) {
```

user-role-editor/includes/settings-template.php, line 133 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 133 (Cross-Site Request Forgery)

Sink:
File: user-role-editor/includes/settings-template.php:133
Taint Flags:

```
130 ?>
131
132  <div id="ure_tabs-2">
133  <form name="ure_additional_modules" method="post" action="<?php echo $link; ?>?page=settings-<?php echo URE_PLUGIN_FILE; ?>" >
134  <table id="ure_addons">
135 <?php
136 if ( ! $multisite ) {
```

user-role-editor/includes/settings-template.php, line 165 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:165
Taint Flags:

```
162 ?>
163
164  <div id="ure_tabs-3">
165  <form name="ure_default_roles" method="post" action="<?php echo $link; ?>?page=settings-<?php echo URE_PLUGIN_FILE; ?>" >
166 <?php
167  if ( ! $multisite ) {
168  esc_html_e( 'Primary default role: ', 'user-role-editor' );
```

user-role-editor/includes/settings-template.php, line 198 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:198
Taint Flags:

```
195 ?>
```


© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 198 (Cross-Site Request Forgery)

```
196  <div id="ure_tabs-4">
197  <div id="ure-settings-form-ms">
198  <form name="ure_settings_ms" method="post" action="<?php echo $link; ?>?page=settings-<?
php echo URE_PLUGIN_FILE; ?>" >
199  <table id="ure_settings_ms">
200 <?php
201  if ( $lib->is_super_admin() ) {
```

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/tools.php, line 18 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/tools.php:18
Taint Flags:

```
15 ?>
16
17  <div style="margin: 10px 0 10px 0; border: 1px solid red; padding: 0 10px 10px 10px; text-
align:left;">
18  <form name="ure_reset_roles_form" id="ure_reset_roles_form" method="post" action="<?php
echo $link; ?>?page=settings-<?php echo URE_PLUGIN_FILE; ?>" >
19  <h3>Reset User Roles</h3>
20  <span style="color: red;"><?php esc_html_e('WARNING!', 'user-role-editor');?></span> 
21 <?php
```

user-role-editor/includes/classes/role-view.php, line 131 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/role-view.php:131
Taint Flags:

```
128  }
129  ob_start();
130 ?>
131  <form name="ure_remove_caps_form" id="ure_remove_caps_form" method="POST"
```


© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/role-view.php, line 131 (Cross-Site Request Forgery)

```
132  action="<?php echo admin_url() . ($network_admin ? 'network/':'') . URE_PARENT .'?
page=users-'.URE_PLUGIN_FILE;?>" >
133  <table id="ure_remove_caps_table">
134  <tr>
```

user-role-editor/includes/classes/role-view.php, line 197 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/role-view.php:197
Taint Flags:

```
194
195 <!-- popup dialogs markup -->
196 <div id="ure_add_role_dialog" class="ure-modal-dialog" style="padding: 10px;">
197  <form id="ure_add_role_form" name="ure_add_role_form" method="POST">
198  <div class="ure-label"><?php esc_html_e('Role name (ID): ', 'user-role-editor'); ?></div>
199  <div class="ure-input"><input type="text" name="user_role_id" id="user_role_id" size="25"/
></div>
200  <div class="ure-label"><?php esc_html_e('Display Role Name: ', 'user-role-editor'); ?></
div>
```

user-role-editor/includes/classes/role-view.php, line 208 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/role-view.php:208
Taint Flags:

```
205 </div>
206
207 <div id="ure_rename_role_dialog" class="ure-modal-dialog" style="padding: 10px;">
208  <form id="ure_rename_role_form" name="ure_rename_role_form" method="POST">
209  <div class="ure-label"><?php esc_html_e('Role name (ID): ', 'user-role-editor'); ?></div>
210  <div class="ure-input"><input type="text" name="ren_user_role_id" id="ren_user_role_id"
size="25" disabled /></div>
211  <div class="ure-label"><?php esc_html_e('Display Role Name: ', 'user-role-editor'); ?></
```


© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/role-view.php, line 208 (Cross-Site Request Forgery)

```
div>
```

user-role-editor/includes/classes/editor.php, line 1493 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/editor.php:1493
Taint Flags:

```
1490  <h1><?php esc_html_e('User Role Editor', 'user-role-editor'); ?></h1>
1491  <div id="ure_container">
1492  <div id="user_role_editor" class="ure-table-cell" >
1493  <form id="ure_form" method="post" action="<?php echo admin_url() . URE_PARENT . '?
page=users-' . URE_PLUGIN_FILE; ?>" >
1494  <div id="ure_form_controls">
1495  <?php
1496  $view->display();
```

Package: user-role-editor.js

user-role-editor/js/ure.js, line 412 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: post
Enclosing Method: role_change()
File: user-role-editor/js/ure.js:412
Taint Flags:

```
409  'sub_action':'get_role_caps',
410  'role': role_id,
411  'wp_nonce': ure_data.wp_nonce};
412  jQuery.post( ajaxurl, data, ure_main.refresh_role_view, 'json' );
413
414  },
415
```


© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: user-role-editor.js

user-role-editor/js/users-grant-roles.js, line 197 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: post
Enclosing Method: ure_revoke_role()
File: user-role-editor/js/users-grant-roles.js:197
Taint Flags:

```
194  'role': role,
195  'wp_nonce': ure_users_grant_roles_data.wp_nonce
196  };
197  jQuery.post(ajaxurl, data, ure_page_reload, 'json');
198
199  return true;
200 }
```

user-role-editor/js/users-grant-roles.js, line 72 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: post
Enclosing Method: ure_get_selected_user_roles()
File: user-role-editor/js/users-grant-roles.js:72
Taint Flags:

```
69  'sub_action':'get_user_roles',
70  'user_id': user_id,
71  'wp_nonce': ure_users_grant_roles_data.wp_nonce};
72  jQuery.post(ajaxurl, data, ure_show_grant_roles_dialog_pre_selected, 'json');
73 }
74
75
```

user-role-editor/js/users-grant-roles.js, line 169 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: post
Enclosing Method: ure_add_role()

© Copyright 2008-2025 Open Text.

Cross-Site Request Forgery — Low

Package: user-role-editor.js

user-role-editor/js/users-grant-roles.js, line 169 (Cross-Site Request Forgery)

File: user-role-editor/js/users-grant-roles.js:169
Taint Flags:

```
166  'role': role,
167  'wp_nonce': ure_users_grant_roles_data.wp_nonce
168  };
169  jQuery.post(ajaxurl, data, ure_page_reload, 'json');
170
171  return true;
172 }
```

user-role-editor/js/users-grant-roles.js, line 141 (Cross-Site Request Forgery)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: post
Enclosing Method: ure_grant_roles()
File: user-role-editor/js/users-grant-roles.js:141
Taint Flags:

```
138  'other_roles': other_roles,
139  'wp_nonce': ure_users_grant_roles_data.wp_nonce
140  };
141  jQuery.post(ajaxurl, data, ure_page_reload, 'json');
142
143  return true;
144 }
```


© Copyright 2008-2025 Open Text.

Cross-Site Scripting: DOM (3 issues)

Abstract

傳送未經驗證的資料至網路瀏覽器，會導致瀏覽器執行惡意的程式碼。

Explanation

Cross-site scripting (XSS) 弱點會在以下情況中出現： 1.資料從一個不可信賴的來源進入 Web 應用程式。在基於 DOM 的 XSS 案例中，會從 URL 參數或瀏覽器內的其他值來讀取資料，並以用戶端程式碼回寫到頁面中。在 Reflected XSS 案例中，不可信賴的來源通常為網頁要求，而在 Persisted XSS (也可稱為 Stored XSS) 案例中，來源通常為資料庫或其他後端資料儲存區。 2.未經驗證且包含在動態內容中的資料將傳送給某個網頁使用者。在基於 DOM 的 XSS 案例中，一旦受害者的瀏覽器開始剖析 HTML 頁面，就會開始執行惡意內容，且成為建立 DOM (Document Object Model，文件物件模型) 的一部分。 傳送到網頁瀏覽器的惡意內容經常是以 JavaScript 片段的形式出現，但是也可能包含 HTML、Flash 或者瀏覽器執行的任何其他程式碼類型。以 XSS 為基礎的攻擊手段花樣百出且幾乎無窮無盡，但是它們通常會傳輸 Cookie 或其他階段作業資訊之類的私人資料給攻擊者、將受害者重新導向到攻擊者控制的網頁內容，或者利用易受攻擊的網站，在使用者的機器上執行其他惡意操作。 **範例 1：**以下的 JavaScript 程式碼片段會從 URL 讀取員工識別碼 `eid` 並顯示給使用者。

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

範例 2：請考慮使用 HTML 表單：

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

以下的 jQuery 程式碼片段會從表單讀取員工識別碼，並顯示給使用者。

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

如果文字輸入中的員工識別碼 (識別碼為 `eid`) 只包含標準英數字元，這些程式碼便會正確地運作。如果 `eid` 中有包含中繼字元或來源程式碼中的值，那麼網路瀏覽器就會像顯示 HTTP 回應那樣執行程式碼。 **範例 3：**以下程式碼顯示了 React 應用程式內基於 DOM 的 XSS 範例：

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

在 `Example 3` 中，如果攻擊者可以控制從 `getUntrustedInput()` 擷取的整個 JSON 物件，則可能會讓 React 將 `element` 解譯為元件，因此可以傳遞具有 `dangerouslySetInnerHTML` 及其自己的控制值的物件 (典型的 Cross-Site Scripting 攻擊)。 一開始，這些似乎不會輕易受到攻擊。畢竟，誰會提供包含在自己電腦上執行的惡意程式碼的輸入？其實，真正的危險在於攻擊者會建立惡意的 URL，接著使用電子郵件或社交工程病毒誘騙受害者透過連結前往該 URL。當受害者按下該連結時，他們即不知不覺地透過易受攻擊的 Web 應用程式，將惡意內容資訊帶回他們自己的電腦。這個利用易受攻擊的 Web 應用程式進行攻擊的機制就是通常所知的 Reflected XSS。 如同範例中所示，XSS 的弱點是由 HTTP 回應中包含未經驗證資料的程式碼所引起的。XSS 攻擊有三種途徑可攻擊受害者： - 直接從 HTTP 要求讀取資料，並直接回傳至 HTTP 回

© Copyright 2008-2025 Open Text.

應。當攻擊者誘使使用者提供危險內容給易受攻擊的 Web 應用程式，接著這些危險內容就會回傳給使用者並由在網路瀏覽器中執行，這時就會出現 Reflected XSS 攻擊行為。傳遞惡意內容最常用的機制就是，將惡意內容當作參數隱藏在公開發表的 URL 中，或者以電子郵件方式直接傳送給受害者。以這種方法建立的 URL 會構成許多網路釣魚 (phishing) 架構的核心，攻擊者可以藉此誘使受害者去造訪一個指向到易受攻擊網站的 URL。網站將攻擊者的內容回傳給使用者之後，就會執行這些內容，並接著從使用者的電腦將可能包含階段作業資訊的 Cookie 之類的私人資訊傳給攻擊者，或者執行其他惡意活動。 - 應用程式會將危險資料儲存到資料庫或其他可信任的資料儲存中。這些危險資料隨後會被回讀到應用程式，並包含在動態內容中。Persistent XSS 攻擊會在以下情況出現：攻擊者把危險內容插入到之後會讀取的資料記憶體中，並包含在動態內容中。從攻擊者的角度來看，插入惡意內容的最佳位置莫過於一個會對很多使用者或特別感興趣的使用者顯示的區域。感興趣的使用者通常會在應用程式中擁有較高的權限，或者會與敏感資料進行互動，且這些資料對攻擊者而言很有利用價值。如果其中一個使用者執行了惡意內容，攻擊者可能會代替使用者去執行需要權限許可的作業，或者取得存取使用者專屬敏感資料的權限。 - 應用程式以外的來源會在資料庫或是其他資料記憶體中儲存危險資料，且之後這些危險資料會被當作信賴的資料回讀到應用程式，並會包含在動態內容中。

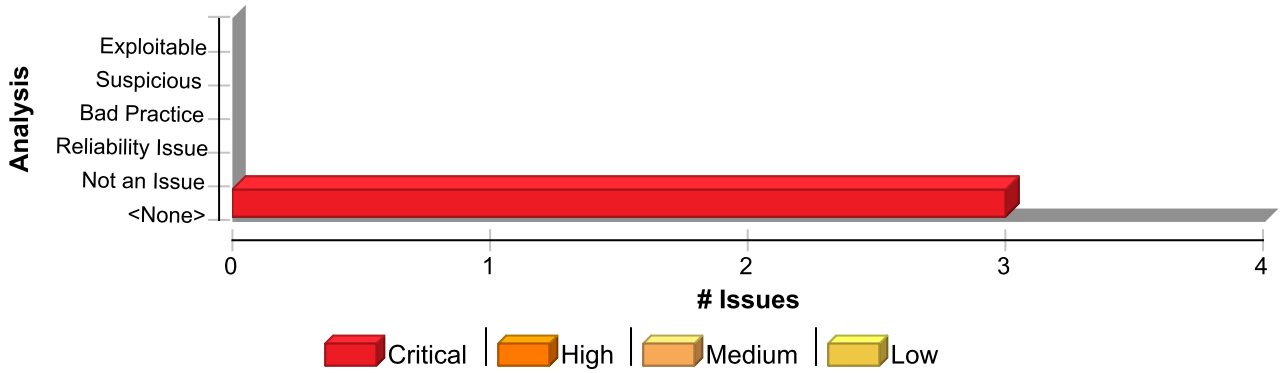
Recommendation

防止 XSS 的解決方法是：確保在需要的位置進行驗證，並設定相關屬性以防止漏洞。 由於 XSS 弱點會在應用程式於輸出中包含惡意資料時出現，所以合理方法就是，在應用程式輸出資料之前馬上驗證資料。然而，由於 Web 應用程式經常會為了產生動態內容而包含複雜且難以理解的程式碼，所以此方法容易遺漏錯誤 (遺漏驗證)。降低此風險的有效方法即為執行 XSS 輸入驗證。 Web 應用程式必須驗證所有輸入以防止出現其他弱點 (如 SQL Injection)，因此比較簡單的方法就是，加強應用程式現有的輸入驗證機制，納入 XSS 檢查。儘管有一定的作用，但是 XSS 的輸入驗證並不能取代嚴格的輸出驗證。應用程式可能會透過共用資料存放區或者其他信賴的來源接受輸入，而該資料存放區可能會從未執行適當輸入驗證的來源接受輸入。因此，應用程式不能間接依賴此資料或任何其他資料的安全性。這代表杜絕 XSS 弱點的最佳方法就是，驗證所有進入應用程式的資料，以及所有離開應用程式並傳送到使用者的資料。 驗證 XSS 最安全的方法，是建立一個安全字元的允許清單，只允許清單上的字元可以出現在 HTTP 內容，並僅接受由經過檢驗的集合中字元組成的輸入。例如，有效的使用者名稱可能僅包含英數字元，或電話號碼可能僅包含數字 0-9。然而，這種解決方式在 Web 應用程式中經常是不可行的，因為很多字元對瀏覽器來說都具有特殊意義，將這些字元編碼之後，必須將它們視為有效輸入。例如，一個網站設計公告欄就必須接受來自於使用者的 HTML 片段。 更具彈性但安全性較低的方法是實作拒絕清單，能在使用輸入前選擇性地拒絕或避開可能有危險的字元。若要建立這類名單，首先必須了解那些對網頁瀏覽器來說有特殊意義的字元集。雖然 HTML 標準會定義具有特殊意義的字元，但是許多網頁瀏覽器會嘗試修正 HTML 中的常見錯誤，而且在特定環境中可能會將其他字元視為有特殊意義。這就是為什麼我們不建議使用拒絕清單做為預防 XSS 的方式。Carnegie Mellon 大學軟體工程學院 (Software Engineering Institute) 下的 CERT(R) 合作中心 (CERT(R) Coordination Center) 提供了在各種環境中具有特殊意義的字元資訊 [1]： 在區塊等級 (block-level) 元素的內容中 (位於文字段落中間)： -「<」是特殊字元，因為它會引入標籤。 -「&」是特殊字元，因為它會引入字元實體。 -「>」是特殊字元，因為某些瀏覽器將其視為特殊字元，會假設該頁面的作者想加入開頭的「<」，卻不小心遺漏掉了。 以下原則適用於屬性值： - 在以雙引號括住的屬性值中，雙引號之所以特殊，是因為它們標記了屬性值的結尾。 - 在以單引號結尾的屬性值中，單引號是特殊字元，因為它們標記了屬性值的結尾。 - 在沒有任何引號的屬性值中，空格字元 (例如空格和定位字元) 也是特殊字元。 -「&」和特定屬性一起使用時會變成特殊字元，因為它會引導出一個字元實體。 舉例來說，在 URL 中，搜索引擎可能會在結果頁面提供一個連結，讓使用者可以按一下該連結來重新執行搜尋。這個方法可以運用在編寫 URL 中的搜尋查詢，此動作會引導出其他特殊字元： - 空格、定位字元和換行符號都是特殊字元，因為它們標記了 URL 的結尾。 -「&」為特殊字元，因為它會引導出一個字元實體或分隔 CGI 參數。 - 非 ASCII 字元 (就是所有在 ISO-8859-1 編碼表中大於 127 的字元) 不允許出現在 URL 中，所以它們在此內容中被視為特殊字元。 - 每當伺服器端程式碼對以 HTTP 逸出序列編碼的參數進行解碼時，就必須從輸入中篩選 "%" 符號。例如，對於諸如「%68%65%6C%6C%6F」的輸入，必須篩選「%」，才能在網頁上顯示「hello」。 在 的正文中： - 將文字直接插入原有指令碼標籤時，必須篩選分號、括號、中括號以及換行字元。 伺服器端 Script： - 如果伺服器端 Script 會將輸入中的任何驚嘆號字元 (!) 轉換為輸出中的雙引號字元時 (")，則可能需要進行更多篩選。 其他可能性： - 若攻擊者提交 UTF-7 編碼的要求，特殊字元「<」會顯示為「+ADw-」，並且可能會避開篩選。如果輸出包含在沒有明確指定編碼格式的頁面中，某些瀏覽器會嘗試以智慧方式來根據內容識別編碼 (在此情況下為 UTF-7)。 在您確定在應用程式中針對 XSS 攻擊執行驗證的正確要點，以及驗證時要考慮的特殊字元後，那麼下一個挑戰就是決定驗證過程中處理特殊字元的方法。如果應用程式將某些特殊字元認定為無效輸入，那麼您可以拒絕任何包含這些被視為無效特殊字元的輸入。第二種選擇是以篩選方式移除特殊字元。然而，篩選所產生的副作用在於會改變篩選內容的顯示樣貌。但是在需要完整顯示輸入內容時，這種情況是無法接受的。 如果必須接受包含特殊字元的輸入且必須準確顯示輸入內容，驗證動作一定要編碼所有特殊字元，以移除特殊字元代表的意義。官方 HTML 規

© Copyright 2008-2025 Open Text.

格 [2] 提供了特殊字元對應的 ISO 8859-1 編碼值完整清單。 很多應用程式伺服器都試圖避免應用程式出現 Cross-site scripting 弱點，方法是為負責設定某個特定 HTTP 回應內容的函數提供各種實作，以驗證是否存在進行 Cross-site scripting 攻擊的字元。請勿依賴執行應用程式的伺服器來確保應用程式的安全性。對於任何開發的應用程式，無法保證在其存留期內將在哪些應用程式伺服器上執行。隨著標準和已知盜取方式不斷地演變，無法保證應用程式伺服器會繼續保持同步。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cross-Site Scripting: DOM	3	0	0	3
Total	**3**	**0**	**0**	**3**

Cross-Site Scripting: DOM — Critical

Package: filebird.includes.Recommended.assets.js

filebird/includes/Recommended/assets/js/recommended.js, line 52 (Cross-Site Scripting: DOM)

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: success(0)
From: success
File: filebird/includes/Recommended/assets/js/recommended.js:49

```
46  tab: tab,
47  nonce: yayRecommended.nonce
48  },
49  success: function(response) {
50  if(response.success === true) {
51  const html = response.data.html;
52  jQuery( ".yay-recommended-plugins-layout #the-list" ).html(html);
```

Sink Details

Sink: ~JS_Generic.html()

© Copyright 2008-2025 Open Text.

Cross-Site Scripting: DOM — Critical

Package: filebird.includes.Recommended.assets.js

filebird/includes/Recommended/assets/js/recommended.js, line 52 (Cross-Site Scripting: DOM)

Enclosing Method: success()
File: filebird/includes/Recommended/assets/js/recommended.js:52
Taint Flags: WEB, XSS

```
49  success: function(response) {
50  if(response.success === true) {
51  const html = response.data.html;
52  jQuery( ".yay-recommended-plugins-layout #the-list" ).html(html);
53  jQuery(".yay-recommended-plugins-layout .plugin-install-tab").removeClass("loading-process");
54  }
55  }
```

Package: user-role-editor.js

user-role-editor/js/users-grant-roles.js, line 235 (Cross-Site Scripting: DOM)

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read href
From: ure_set_url_arg
File: user-role-editor/js/users-grant-roles.js:204

```
201
202
203 function ure_set_url_arg(arg_name, arg_value) {
204  var url = window.location.href;
205  var hash = location.hash;
206  url = url.replace(hash, '');
207  if (url.indexOf(arg_name + "=")>=0) {
```

Sink Details

Sink: Assignment to document.location
Enclosing Method: ure_page_reload()
File: user-role-editor/js/users-grant-roles.js:235
Taint Flags: CONCATENATED, VALIDATED_OPEN_REDIRECT, WEB, XSS

```
232  }
233
234  var url = ure_set_url_arg('update', 'promote');
235  document.location = url;
236
237 }
```


© Copyright 2008-2025 Open Text.

Cross-Site Scripting: DOM — Critical

Package: user-role-editor.js

user-role-editor/js/users-grant-roles.js, line 235 (Cross-Site Scripting: DOM)

```
238
```

user-role-editor/js/users-grant-roles.js, line 235 (Cross-Site Scripting: DOM)

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read location.hash
From: ure_set_url_arg
File: user-role-editor/js/users-grant-roles.js:205

```
202
203 function ure_set_url_arg(arg_name, arg_value) {
204  var url = window.location.href;
205  var hash = location.hash;
206  url = url.replace(hash, '');
207  if (url.indexOf(arg_name + "=")>=0) {
208  var prefix = url.substring(0, url.indexOf(arg_name));
```

Sink Details

Sink: Assignment to document.location
Enclosing Method: ure_page_reload()
File: user-role-editor/js/users-grant-roles.js:235
Taint Flags: CONCATENATED, VALIDATED_OPEN_REDIRECT, WEB, XSS

```
232  }
233
234  var url = ure_set_url_arg('update', 'promote');
235  document.location = url;
236
237 }
238
```


© Copyright 2008-2025 Open Text.

Hidden Field (13 issues)

Abstract

已使用隱藏表單欄位。

Explanation

程式設計師通常會信任隱藏欄位的內容，預期使用者無法檢視或竄改他們的內容。攻擊者將會推翻這些假設。他們會檢查寫入隱藏欄位的值並加以修改，或以攻擊資料取代這些內容。 **範例 1：**hidden 類型的 標籤表示使用隱藏欄位。

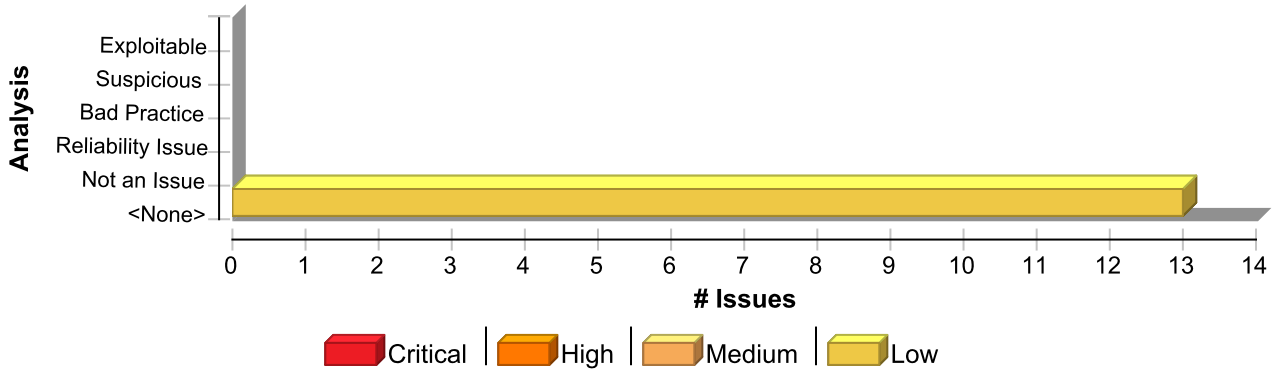
```
<input type="hidden">
```

若隱藏欄位具有敏感資訊，該資訊的快取方式將與其他的頁面相同。這可能會導致敏感資訊在使用者不知情的狀況下貯存至瀏覽器快取中。

Recommendation

預期攻擊者會檢閱並解碼應用程式中所有隱藏欄位的使用。請將隱藏欄位視為不可信賴的輸入。若特定資訊不應與其他頁面一同快取，請勿將其儲存到隱藏欄位中。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Hidden Field	13	0	0	13
Total	**13**	**0**	**0**	**13**

Hidden Field — Low

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 119 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:119
Taint Flags:

© Copyright 2008-2025 Open Text.

Hidden Field — Low

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 119 (Hidden Field)

```
116 ?>
117  </table>
118  <?php wp_nonce_field( 'user-role-editor' ); ?>
119  <input type="hidden" name="ure_tab_idx" value="0" />
120  <p class="submit">
121  <input type="submit" class="button-primary" name="ure_settings_update" value="<?php
esc_html_e( 'Save', 'user-role-editor' ) ?>" />
122  </p>
```

user-role-editor/includes/settings-template.php, line 154 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:154
Taint Flags:

```
151 ?>
152  </table>
153 <?php wp_nonce_field( 'user-role-editor' ); ?>
154  <input type="hidden" name="ure_tab_idx" value="1" />
155  <p class="submit">
156  <input type="submit" class="button-primary" name="ure_addons_settings_update" value="<?
php esc_html_e('Save', 'user-role-editor') ?>" />
157
```

user-role-editor/includes/settings-template.php, line 186 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:186
Taint Flags:

```
183 ?>
184  <hr>
185  <?php wp_nonce_field( 'user-role-editor' ); ?>
186  <input type="hidden" name="ure_tab_idx" value="<?php echo $tabs_index[3]; ?>" />
187  <p class="submit">
188  <input type="submit" class="button-primary" name="ure_default_roles_update" value="<?php
esc_html_e( 'Save', 'user-role-editor' ) ?>" />
```


© Copyright 2008-2025 Open Text.

Hidden Field — Low

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 186 (Hidden Field)

```
189  </p>
```

user-role-editor/includes/settings-template.php, line 218 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:218
Taint Flags:

```
215 ?>
216  </table>
217 <?php wp_nonce_field( 'user-role-editor' ); ?>
218  <input type="hidden" name="ure_tab_idx" value="<?php echo $tabs_index[4]; ?>" />
219  <p class="submit">
220  <input type="submit" class="button-primary" name="ure_settings_ms_update" value="<?php
esc_html_e( 'Save', 'user-role-editor' ); ?>" />
221  </p>
```

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/editor.php, line 1499 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/editor.php:1499
Taint Flags:

```
1496  $view->display();
1497  wp_nonce_field( 'user-role-editor', 'ure_nonce' );
1498  ?>
1499  <input type="hidden" name="action" value="update" />
1500  </div>
1501  </form>
1502 <?php
```

user-role-editor/includes/classes/user-view.php, line 205 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

© Copyright 2008-2025 Open Text.

Hidden Field — Low

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/user-view.php, line 205 (Hidden Field)

Sink Details

Sink:
File: user-role-editor/includes/classes/user-view.php:205
Taint Flags:

```
202  </td>
203  </tr>
204 </table>
205  <input type="hidden" name="object" value="user" />
206  <input type="hidden" name="user_id" value="<?php echo $this->user_to_edit->ID; ?>" />
207  </div>
208 </div>
```

user-role-editor/includes/classes/user-view.php, line 206 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/user-view.php:206
Taint Flags:

```
203  </tr>
204 </table>
205  <input type="hidden" name="object" value="user" />
206  <input type="hidden" name="user_id" value="<?php echo $this->user_to_edit->ID; ?>" />
207  </div>
208 </div>
209 <?php
```

user-role-editor/includes/classes/tools.php, line 42 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/tools.php:42
Taint Flags:

```
39  <br><br>
40  <button id="ure_reset_roles_button" style="width: 100px; color: red;" title="<?php
esc_html_e('Reset Roles to its original state', 'user-role-editor'); ?>" disabled><?php
esc_html_e('Reset', 'user-role-editor');?></button>
```


© Copyright 2008-2025 Open Text.

Hidden Field — Low

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/tools.php, line 42 (Hidden Field)

```
41  <?php wp_nonce_field('user-role-editor'); ?>
42  <input type="hidden" name="ure_settings_tools_exec" value="1" />
43  <input type="hidden" name="ure_reset_roles_exec" value="1" />
44  <input type="hidden" name="ure_tab_idx" value="<?php echo $tab_idx; ?>" />
45  </form>
```

user-role-editor/includes/classes/tools.php, line 43 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/tools.php:43
Taint Flags:

```
40  <button id="ure_reset_roles_button" style="width: 100px; color: red;" title="<?php
esc_html_e('Reset Roles to its original state', 'user-role-editor'); ?>" disabled><?php
esc_html_e('Reset', 'user-role-editor');?></button>
41  <?php wp_nonce_field('user-role-editor'); ?>
42  <input type="hidden" name="ure_settings_tools_exec" value="1" />
43  <input type="hidden" name="ure_reset_roles_exec" value="1" />
44  <input type="hidden" name="ure_tab_idx" value="<?php echo $tab_idx; ?>" />
45  </form>
46  </div>
```

user-role-editor/includes/classes/tools.php, line 44 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/tools.php:44
Taint Flags:

```
41  <?php wp_nonce_field('user-role-editor'); ?>
42  <input type="hidden" name="ure_settings_tools_exec" value="1" />
43  <input type="hidden" name="ure_reset_roles_exec" value="1" />
44  <input type="hidden" name="ure_tab_idx" value="<?php echo $tab_idx; ?>" />
45  </form>
46  </div>
47
```


© Copyright 2008-2025 Open Text.

Hidden Field — Low

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/role-view.php, line 157 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/role-view.php:157
Taint Flags:

```
154  } // foreach($caps...)
155 ?>
156  </table>
157  <input type="hidden" name="action" id="action" value="delete-user-capability" />
158  <input type="hidden" name="user_role" id="ure_role" value="<?php echo $current_role;?>" />
159  <?php wp_nonce_field('user-role-editor', 'ure_nonce'); ?>
160  </form>
```

user-role-editor/includes/classes/role-view.php, line 158 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/role-view.php:158
Taint Flags:

```
155 ?>
156  </table>
157  <input type="hidden" name="action" id="action" value="delete-user-capability" />
158  <input type="hidden" name="user_role" id="ure_role" value="<?php echo $current_role;?>" />
159  <?php wp_nonce_field('user-role-editor', 'ure_nonce'); ?>
160  </form>
161 <?php
```

user-role-editor/includes/classes/role-view.php, line 409 (Hidden Field)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/role-view.php:409
Taint Flags:

© Copyright 2008-2025 Open Text.

Hidden Field — Low

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/role-view.php, line 409 (Hidden Field)

```
406  $current_role = $this->editor->get('current_role');
407  $ao->show($current_role);
408 ?>
409  <input type="hidden" name="object" value="role" />
410  </div>
411  </div>
412 <?php
```


© Copyright 2008-2025 Open Text.

Insecure Randomness (2 issues)

Abstract

標準的虛擬亂數產生器也無法抵擋加密攻擊。

Explanation

在安全性要求較高的環境中，將一個能夠產生可預測值的函數當作隨機來源使用時，會產生 Insecure Randomness 錯誤。 電腦是一種決定性的機器，因此不可能產生真正的隨機性。虛擬亂數產生器 (PRNG, Pseudorandom Number Generator) 近似於隨機演算，會從一個可以計算後續值的種子開始。 PRNG 包括兩種類型：統計式和加密式。統計式 PRNG 可提供有用的統計內容，但是它們的輸出很容易預測，導致複製數值串流很容易。因此，對於因安全性由產生數值決定而導致其不可預測的環境，其並不適用。加密式 PRNG 則會藉由產生更難以預測的輸出來解決這個問題。若要對數值進行加密保護，必須使攻擊者無法或難以區別產生的隨機數值和真實的隨機數值。一般來說，如果沒有指出某個 PRNG 演算法經過加密保護，那麼它很可能是統計式 PRNG，不應在安全性要求較高的環境中使用，否則會引發嚴重弱點，例如易猜的臨時密碼、容易預測的加密金鑰、Session Hijacking 以及 DNS 欺騙。 **範例 1：**以下程式碼使用統計式 PRNG，為購買後仍在有效期內的收據建立 URL。

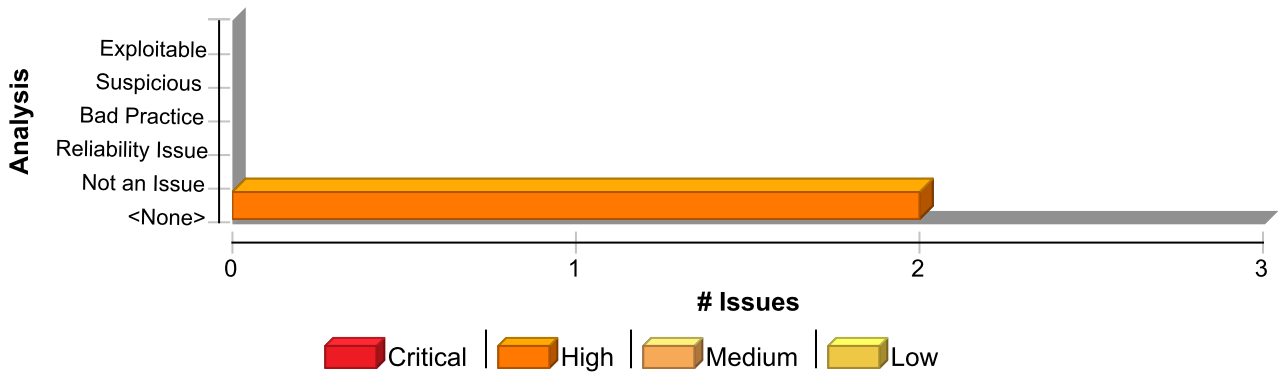
```
function genReceiptURL (baseURL){
var randNum = Math.random();
var receiptURL = baseURL + randNum + ".html";
return receiptURL;
}
```

此程式碼使用 `Math.random()` 函數為其所產生的收據頁面產生「唯一」識別碼。由於 `Math.random()` 是統計式 PRNG，攻擊者很容易就能猜到其所產生的字串。雖然收據系統的基礎設計也存在錯誤，但是如果它使用一個不會產生可預測收據識別碼的亂數產生器 (例如加密式 PRNG)，那就會安全很多。

Recommendation

當不可預測性至關重要時 (例如大多數對安全性要求較高的環境都採用隨機性)，請使用加密型 PRNG。不管您選擇哪一種 PRNG，請務必使用擁有足夠複雜度 (entropy) 的值來進行演算。(像當前值的複雜度就很低，因此不宜使用。) 對於 Node.js 應用程式，請考慮使用 `crypto` 模組中的函數，以提供加密安全的隨機數。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Insecure Randomness	2	0	0	2
Total	**2**	**0**	**0**	**2**

© Copyright 2008-2025 Open Text.

Insecure Randomness — High

Package: user-role-editor.js

user-role-editor/js/multiple-select.js, line 315 (Insecure Randomness)

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: random
Enclosing Method: lambda()
File: user-role-editor/js/multiple-select.js:315
Taint Flags:

```
312  var nativeWeakMap = typeof WeakMap === 'function' && /native
code/.test(functionToString.call(WeakMap));
313
314  var id = 0;
315  var postfix = Math.random();
316
317  var uid = function (key) {
318  return 'Symbol(' + String(key === undefined ? '' : key) + ')_' + (++id +
postfix).toString(36);
```

user-role-editor/js/multiple-select.js, line 3012 (Insecure Randomness)

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionPointerCall: random
Enclosing Method: getDocumentClickEvent()
File: user-role-editor/js/multiple-select.js:3012
Taint Flags:

```
3009
3010  var getDocumentClickEvent = function getDocumentClickEvent() {
3011  var id = arguments.length > 0 && arguments[0] !== undefined ? arguments[0] : '';
3012  id = id || "".concat(+new Date()).concat(~~(Math.random() * 1000000));
3013  return "click.multiple-select-".concat(id);
3014  };
3015
```


© Copyright 2008-2025 Open Text.

Insecure Transport: External Link (3 issues)

Abstract

檔案透過未加密的通道連結到第三方網站。

Explanation

請確保網頁上的超連結僅連結到安全位置，以防止瀏覽網站時發生任何使用者洩漏資訊。即使連結從不安全的通訊協定 (例如 HTTP) 重新導向到安全的通訊協定 (例如 HTTPS)，透過未加密通道的初始連線也會讓攻擊者能夠執行 Man-in-the-Middle (MiTM) 攻擊。如此一來，攻擊者就能夠控制頁面，即產生的登陸頁面。 **範例 1：**請考慮以下超連結：

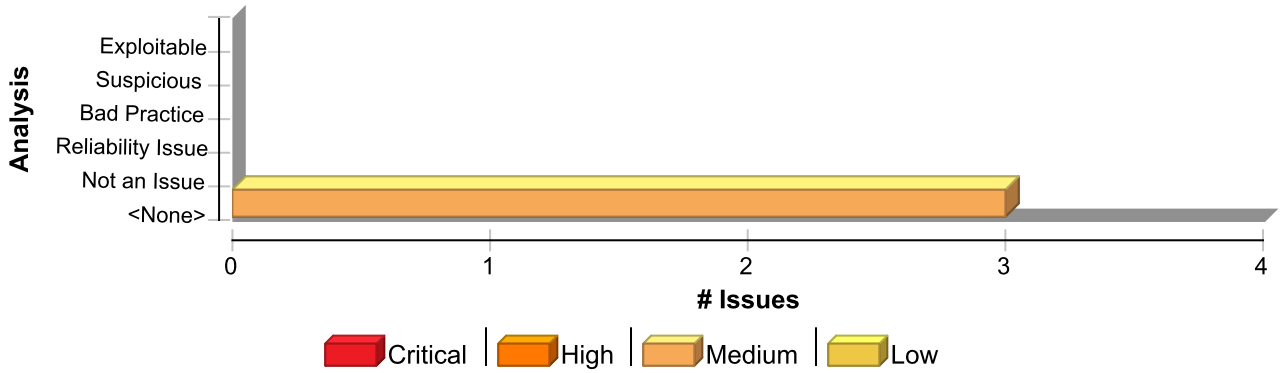
```
<a href="http://www.example.com/index.html"/>
```

如果攻擊者正在監聽使用者與伺服器之間的網路流量，就可以模擬或操縱 `www.example.com` 來載入自己的網頁。第三方網站的連結最初可能不被認為對安全很重要，但是對使用者來說，任何洩漏都可能顯示為來自您網頁上的連結，因此會降低使用者對使用您平台的信任度。

Recommendation

請控制您網站上連結的網頁，並儘可能確保始終透過安全的通訊協定載入連結。如果目標伺服器需要使用不安全的通訊協定，請提供警語，告知使用者按一下連結會有額外風險。請儘可能不要使用來自第三方網站的 script 或其他人工項目。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Insecure Transport: External Link	3	0	0	3
Total	**3**	**0**	**0**	**3**

Insecure Transport: External Link — Medium

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 13 (Insecure Transport: External Link)

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Content)

© Copyright 2008-2025 Open Text.

Insecure Transport: External Link — Medium

Package: user-role-editor.includes

user-role-editor/includes/settings-template.php, line 13 (Insecure Transport: External Link)

Sink Details

Sink:
File: user-role-editor/includes/settings-template.php:13
Taint Flags:

```
10 $tabs_index = array();
11 ?>
12 <div class="wrap">
13  <a href="http://role-editor.com">
14  <div id="ure-icon" class="icon32"><br></div>
15  </a>
16  <h1><?php esc_html_e( 'User Role Editor - Options', 'user-role-editor' ); ?></h1>
```

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/lib.php, line 389 (Insecure Transport: External Link)

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/lib.php:389
Taint Flags:

```
386  <h2>User Role Editor</h2>
387
388  <strong><?php esc_html_e('Version:', 'user-role-editor');?></strong> <?php echo
URE_VERSION; ?><br/><br/>
389  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/
vladimir.png'; ?>);" target="_blank" href="http://www.shinephp.com/"><?php
esc_html_e("Author's website", 'user-role-editor'); ?></a><br/>
390  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/
user-role-editor-icon.png'; ?>);" target="_blank" href="https://www.role-editor.com"><?php
esc_html_e('Plugin webpage', 'user-role-editor'); ?></a><br/>
391  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/
user-role-editor-icon.png'; ?>);" target="_blank" href="https://www.role-editor.com/download-
plugin"><?php esc_html_e('Plugin download', 'user-role-editor'); ?></a><br/>
392  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/
changelog-icon.png'; ?>);" target="_blank" href="https://www.role-editor.com/changelog"><?php
esc_html_e('Changelog', 'user-role-editor'); ?></a><br/>
```

user-role-editor/includes/classes/lib.php, line 393 (Insecure Transport: External Link)

Issue Details

Kingdom: Security Features

© Copyright 2008-2025 Open Text.

Insecure Transport: External Link — Medium

Package: user-role-editor.includes.classes

user-role-editor/includes/classes/lib.php, line 393 (Insecure Transport: External Link)

Scan Engine: SCA (Content)

Sink Details

Sink:
File: user-role-editor/includes/classes/lib.php:393
Taint Flags:

```
390  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/user-role-editor-icon.png'; ?>);" target="_blank" href="https://www.role-editor.com"><?php esc_html_e('Plugin webpage', 'user-role-editor'); ?></a><br/>
391  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/user-role-editor-icon.png'; ?>);" target="_blank" href="https://www.role-editor.com/download-plugin"><?php esc_html_e('Plugin download', 'user-role-editor'); ?></a><br/>
392  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/changelog-icon.png'; ?>);" target="_blank" href="https://www.role-editor.com/changelog"><?php esc_html_e('Changelog', 'user-role-editor'); ?></a><br/>
393  <a class="ure_rsb_link" style="background-image:url(<?php echo URE_PLUGIN_URL . 'images/faq-icon.png'; ?>);" target="_blank" href="http://www.shinephp.com/user-role-editor-wordpress-plugin/#faq"><?php esc_html_e('FAQ', 'user-role-editor'); ?></a><br/>
394 <?php
395  }
396  // end of about()
```


© Copyright 2008-2025 Open Text.

JSON Injection (1 issue)

Abstract

此方法將未經驗證的輸入寫入 JSON。此呼叫可讓攻擊者將任意元素或屬性注入 JSON 實體。

Explanation

JSON Injection 發生於： 1. 資料從一個不可信賴的來源進入程式。 2. 資料會寫入 JSON 串流。 應用程式通常使用 JSON 來儲存資料或傳送訊息。JSON 用於儲存資料時，通常視為快取資料，可能會包含敏感資訊。JSON 用於傳送訊息時，通常會與 RESTful 服務搭配使用，並且可以用於傳輸敏感資訊，例如驗證憑證。 如果應用程式從未驗證的輸入建構 JSON，則 JSON 文件和訊息的語義便可能會更改。在相對不具惡意的情況下，攻擊者可能會注入外來的元素，而造成應用程式在剖析 JSON 文件或要求時拋出一個異常。在更嚴重的情況下，例如涉及 JSON Injection，攻擊者可能會插入外來元素，而允許對 JSON 文件或要求內的營運關鍵值執行可預測的操作。在某些情況下，JSON Injection 可能引發 Cross-site scripting 或 Dynamic code 評估。 **範例 1：**以下 JavaScript 程式碼使用 jQuery 在值來自 URL 的位置中剖析 JSON：

```
var str = document.URL;
var url_check = str.indexOf('name=');
var name = null;
if (url_check > -1) {
name =  decodeURIComponent(str.substring((url_check+5), str.length));
}

$(document).ready(function(){
if (name !== null){
var obj = jQuery.parseJSON('{"role": "user", "name" : "' + name + '"}');
...
}
...
});
```

此處將不會驗證 `name` 中不可信賴的資料，以逸出與 JSON 相關的特殊字元。如此便允許使用者任意注入 JSON 金鑰，進而可能變更已序列化之 JSON 的結構。在此範例中，如果非特殊權限使用者 `mallory` 將 `","role":"admin` 附加至 URL 中的名稱參數，JSON 會變成：

```
{
"role":"user",
"username":"mallory",
"role":"admin"
}
```

這會使用 `jQuery.parseJSON()` 進行剖析並設定為純物件，這表示 `obj.role` 現在會傳回 "admin"，而不是 "user"

Recommendation

將使用者提供的資料寫入 JSON 時，必須遵循以下原則： 1.請勿使用擷取自使用者輸入的名稱來建立 JSON 屬性。 2. 請務必一律使用安全的序列化函數來執行序列化為 JSON，安全的序列化函數會隔開單引號或雙引號中的不可信賴資料，並逸出任何特殊字元。 **範例 2：**以下 JavaScript 程式碼實作與 `Example 1` 中相同的功能，但改為比對安全名單來驗證名稱並拒絕值，否則會在剖析 JSON 之前將名稱設定為「guest」：

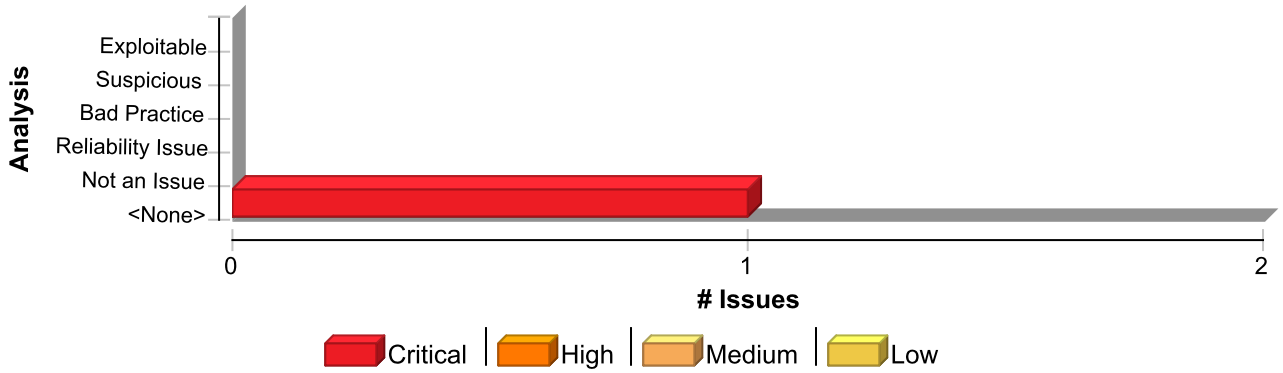
```
var str = document.URL;
var url_check = str.indexOf('name=');
var name = null;
if (url_check > -1) {
name =  decodeURIComponent(str.substring((url_check+5), str.length));
}

function getName(name){
var regexp = /^[A-z0-9]+$/;
```


© Copyright 2008-2025 Open Text.

```
var matches = name.match(regexp);
if (matches == null){
return "guest";
} else {
return name;
}
}

$(document).ready(function(){
if (name !== null){
var obj = jQuery.parseJSON('{"role": "user", "name" : "' + getName(name) +
'"}');
...
}
...
});
```

雖然在此案例中，可以用此方式執行允許清單，但是因為我們希望使用者控制名稱，所以在其他案例中，最好使用使用者完全無法控制的值。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
JSON Injection	1	0	0	1
Total	**1**	**0**	**0**	**1**

JSON Injection — Critical

Package: user-role-editor.js

user-role-editor/js/users.js, line 66 (JSON Injection)

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: success(0)
From: success
File: user-role-editor/js/users.js:65

```
62  wp_nonce: ure_users_data.wp_nonce,
```


© Copyright 2008-2025 Open Text.

JSON Injection — Critical

Package: user-role-editor.js

user-role-editor/js/users.js, line 66 (JSON Injection)

```
63  new_role: new_role
64  },
65  success: function(response) {
66  var data = jQuery.parseJSON(response);
67  if (typeof data.result !== 'undefined') {
68  if (data.result === 'success') {
```

Sink Details

Sink: parseJSON()
Enclosing Method: success()
File: user-role-editor/js/users.js:66
Taint Flags: WEB, XSS

```
63  new_role: new_role
64  },
65  success: function(response) {
66  var data = jQuery.parseJSON(response);
67  if (typeof data.result !== 'undefined') {
68  if (data.result === 'success') {
69  ure_post_move_users_command(data);
```


© Copyright 2008-2025 Open Text.

Often Misused: File Upload (1 issue)

Abstract

允許使用者上傳檔案可能會使攻擊者在伺服器執行已插入的危險內容或惡意程式碼。

Explanation

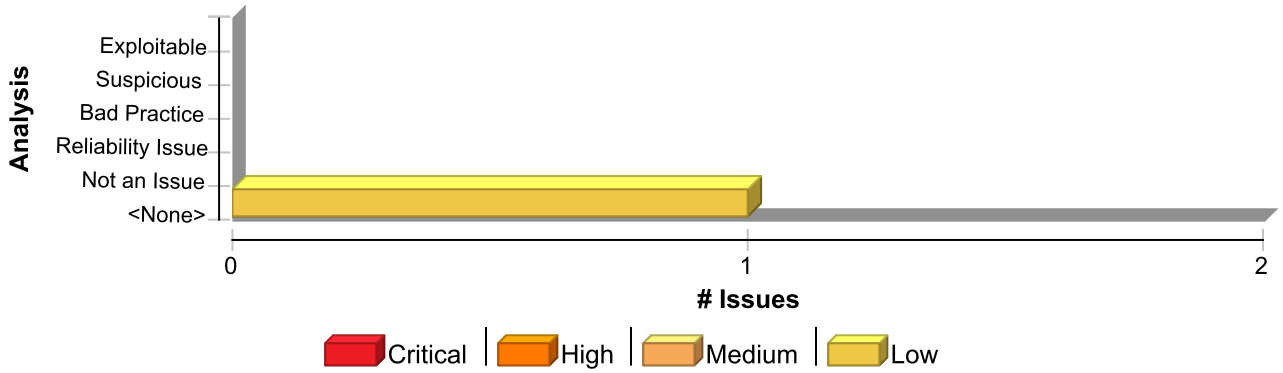
不論撰寫程式所使用的語言為何，毀壞性最大的攻擊通常都包含了遠端程式碼的執行，攻擊者可藉此在程式的上下文環境中成功的執行惡意程式碼。若攻擊者能夠上傳檔案至一個可從網路存取的目錄，並將這些檔案傳遞至程式碼解譯器 (例如 JSP/ASPX/PHP)，則他們可能會造成這些檔案內包含的惡意程式碼在伺服器上執行。即使程式將上傳的檔案儲存於無法從 Web 存取的目錄下，攻擊者仍然可能透過向伺服器環境引入惡意內容中來發動其他攻擊。若程式容易有 Path Manipulation、Command Injection 或 Dangerous File Inclusion 的弱點，則攻擊者可能會上傳含有惡意內容的檔案並利用另一項弱點，使程式讀取或執行該檔。 `file` 類型的 標籤表示程式接受檔案上傳。 **範例 1：**

```
<input type="file">
```

Recommendation

在可以避免的情況下，請勿允許檔案上傳。如果程式必須接受上傳的檔案，則應只接受程式預期的特定內容類型，阻止攻擊者提供惡意內容。大部分依賴上傳內容的攻擊都需要攻擊者能夠提供他們所選擇的內容。限制程式所接受的內容可大幅減少發生攻擊的可能性。請檢查檔案名稱、副檔名及檔案內容，以確定他們皆符合預期並可供應用程式使用。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Often Misused: File Upload	1	0	0	1
Total	**1**	**0**	**0**	**1**

Often Misused: File Upload — Low

Package: filebird.views.pages.settings

filebird/views/pages/settings/tab-import.php, line 221 (Often Misused: File Upload)

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Content)

© Copyright 2008-2025 Open Text.

Often Misused: File Upload — Low

Package: filebird.views.pages.settings

filebird/views/pages/settings/tab-import.php, line 221 (Often Misused: File Upload)

Sink Details

Sink:
File: filebird/views/pages/settings/tab-import.php:221
Taint Flags:

```
218  </th>
219  <td>
220  <div class="flex-item-center">
221  <input type="file" accept=".csv" id="njt-fb-upload-csv" name="csv_file">
222  </div>
223  <p class="hidden">
224  Choose user folder:
```


© Copyright 2008-2025 Open Text.

Possible Variable Overwrite: Global Scope (2 issues)

Abstract

這將會覆寫全域變數，但也等於是為攻擊者製造機會。

Explanation

函數若可覆寫已初始化的全域變數，則也會讓攻擊者得以影響依賴已覆寫變數的程式碼執行。 **範例 1：**若攻擊者在以下 PHP 程式碼區段中為 `str` 提供惡意值，則呼叫 `mb_parse_str()` 時可能會覆寫包括 `first` 在內的任意變數。在此案例中，若包含 JavaScript 的惡意值覆寫 `first`，此程式會容易受到 Cross-Site Scripting 攻擊。

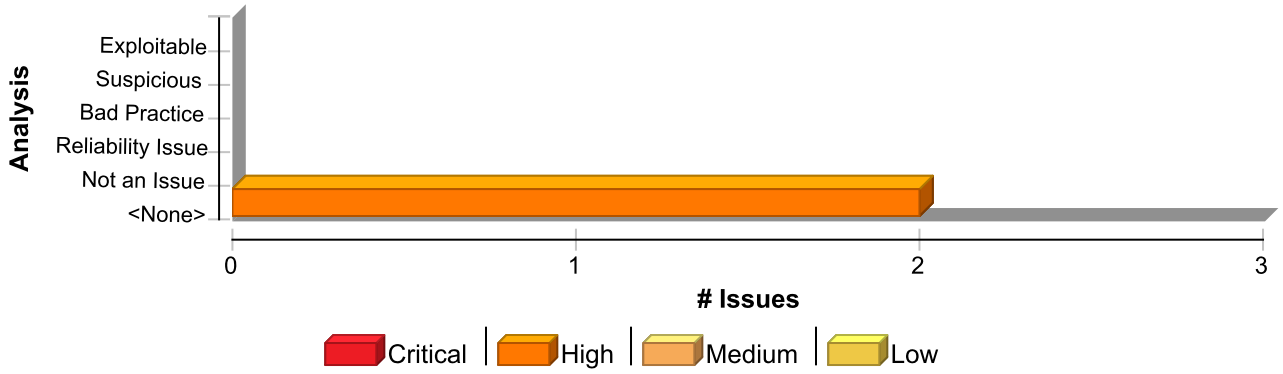
```
<?php
$first="User";
...
$str =  $_SERVER['QUERY_STRING'];
mb_parse_str($str);
echo $first;
?>
```

Recommendation

利用以下方式，防止可覆寫全域變數的函數執行上述操作： - 使用第二個引數呼叫 `mb_parse_str(string $encoded_string [, array &$result ])` 以取得作業的結果，並防止函數覆寫全域變數。 - 將第二個引數設成 EXTR_SKIP，呼叫 `extract(array $var_array [, int $extract_type [, string $prefix]])` 以防止函數覆寫已定義的全域變數。 **範例 2：**以下程式碼使用 `mb_parse_str()` 的第二個引數以減少 `Example 1` 中的弱點。

```
<?php
$first="User";
...
$str =  $_SERVER['QUERY_STRING'];
mb_parse_str($str, $output);
echo $first;
?>
```

Issue Summary



© Copyright 2008-2025 Open Text.

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Possible Variable Overwrite: Global Scope	2	0	0	2
Total	**2**	**0**	**0**	**2**

Possible Variable Overwrite: Global Scope — High

Package: filebird~^~classes

filebird/includes/Classes/Helpers.php, line 203 (Possible Variable Overwrite: Global Scope)

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: view()
File: filebird/includes/Classes/Helpers.php:203
Taint Flags:

```
200  }
201
202  public static function view( $path, $data = array() ) {
203  extract( $data );
204  ob_start();
205  include_once NJFB_PLUGIN_PATH . 'views/' . $path . '.php';
206  return ob_get_clean();
```

Package: filebird~^~controller

filebird/includes/Controller/Controller.php, line 20 (Possible Variable Overwrite: Global Scope)

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: extract
Enclosing Method: loadview()
File: filebird/includes/Controller/Controller.php:20
Taint Flags:

```
17  if ( ! file_exists( $viewPath ) ) {
18  die( 'View <strong>' . esc_html( $viewPath ) . '</strong> not found!' );
19  }
20  extract( $data );
21  if ( $return_html === true ) {
22  ob_start();
23  include_once $viewPath;
```


© Copyright 2008-2025 Open Text.

System Information Leak: Internal (1 issue)

Abstract

顯示系統資料或除錯資訊可讓攻擊者使用系統資訊制訂攻擊計畫。

Explanation

在透過列印或記錄將系統資料或除錯資訊傳送至本機檔案、主控台或螢幕時，會發生內部資訊洩漏。 **範例 1：**以下程式碼會將異常寫入標準錯誤串流：

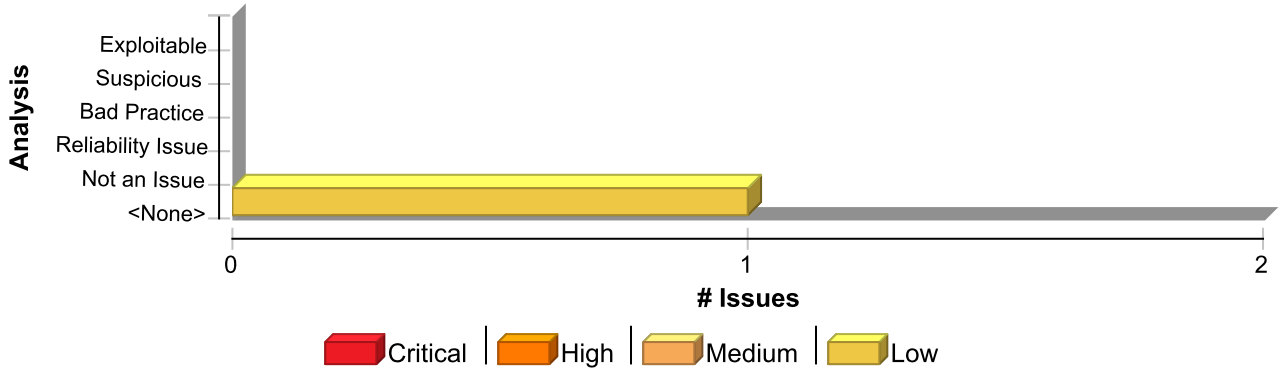
```
var http = require('http');
...

http.request(options, function(res){
...
}).on('error', function(e){
console.log('There was a problem with the request: ' + e);
});
...
```

根據系統配置，可以將資訊傾印到主控台，寫入記錄檔，或者顯示給使用者。在某些情況下，錯誤訊息會為攻擊者提供系統容易受到哪些確切的攻擊類型影響。例如，資料庫錯誤訊息即可揭露應用程式容易受到 SQL injection 攻擊。其他錯誤訊息還可揭露更多關於系統的間接線索。在 `Example 1` 中，洩漏的資訊可能會暗示有關作業系統類型、系統上安裝的應用程式，以及管理員在配置程式時所花費的努力等資訊。

Recommendation

編寫錯誤訊息時，請將安全性問題考慮進去。在生產環境中，請儘量使用簡短的錯誤訊息，而不要使用詳細的錯誤資訊。限制產生與儲存詳細的輸出內容，將有助於管理員和程式設計師診斷問題。除錯追蹤有時可能會出現在不明顯的地方 (例如，內嵌在錯誤頁面的 HTML 註解中)。 即使是不會揭露堆疊追蹤或資料庫傾印的簡短錯誤訊息，都可能會助攻擊者一臂之力。例如，「拒絕存取」訊息可能表示系統中存在著檔案或使用者。

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
System Information Leak: Internal	1	0	0	1
Total	**1**	**0**	**0**	**1**

© Copyright 2008-2025 Open Text.

System Information Leak: Internal — Low

Package: filebird.assets.js

filebird/assets/js/setting.js, line 639 (System Information Leak: Internal)

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: lambda(0)
From: lambda
File: filebird/assets/js/setting.js:639

```
636   .then((res) => {
637   jQuery(this).next().text(res.message);
638   })
639   .catch((err) => console.log(err))
640   .always(() => {
641   jQuery(this).removeClass("updating-message");
642   });
```

Sink Details

Sink: ~JS_Generic.log()
Enclosing Method: lambda()
File: filebird/assets/js/setting.js:639
Taint Flags: SYSTEMINFO

```
636   .then((res) => {
637   jQuery(this).next().text(res.message);
638   })
639   .catch((err) => console.log(err))
640   .always(() => {
641   jQuery(this).removeClass("updating-message");
642   });
```


© Copyright 2008-2025 Open Text.

Description of Key Terminology

Likelihood and Impact

Likelihood
Likelihood is the probability that a vulnerability will be accurately identified and successfully exploited.

Impact
Impact is the potential damage an attacker could do to assets by successfully exploiting a vulnerability. This damage can be in the form of, but not limited to, financial loss, compliance violation, loss of brand reputation, and negative publicity.

Fortify Priority Order

Critical
Critical-priority issues have high impact and high likelihood. Critical-priority issues are easy to detect and exploit and result in large asset damage. These issues represent the highest security risk to the application. As such, they should be remediated immediately.

SQL Injection is an example of a critical issue.

High
High-priority issues have high impact and low likelihood. High-priority issues are often difficult to detect and exploit, but can result in large asset damage. These issues represent a high security risk to the application. High-priority issues should be remediated in the next scheduled patch release.

Password Management: Hardcoded Password is an example of a high issue.

Medium
Medium-priority issues have low impact and high likelihood. Medium-priority issues are easy to detect and exploit, but typically result in small asset damage. These issues represent a moderate security risk to the application. Medium-priority issues should be remediated in the next scheduled product update.

Path Manipulation is an example of a medium issue.

Low
Low-priority issues have low impact and low likelihood. Low-priority issues can be difficult to detect and exploit and typically result in small asset damage. These issues represent a minor security risk to the application. Low-priority issues should be remediated as time allows.

Dead Code is an example of a low issue.

© Copyright 2008-2025 Open Text.

About Fortify Solutions

Fortify is the leader in end-to-end application security solutions with the flexibility of testing on-premise and on-demand to cover the entire software development lifecycle. Learn more at www.microfocus.com/solutions/application-security.

© Copyright 2008-2025 Open Text.