cicdform_backend 掃描報告

專案名稱	cicdform_backend
掃描開始	2024年6月26日 下午 02:22:59
預設集合	Checkmarx Default
掃描時間	00h:00m:49s
被掃描的程式行數	10790
被掃描的檔案數	155
報告建立時間	2024年6月26日 下午 02:27:45
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446
團隊	planning
Checkmarx版本	9.5.5.1007 HF14
掃描類別	完整的
來源	LocalPath
漏洞密度	2/100 (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: 校驗, 不可利用, 確認, 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 v3.2.1	全部
OWASP Top 10 2013	全部
FISMA 2014	全部
NIST SP 800-53	全部
OWASP Top 10 2017	全部
OWASP Mobile Top 10 2016	全部
OWASP Top 10 API	全部
ASD STIG 4.10	全部
OWASP Top 10 2010	全部
CWE top 25	全部
MOIS(KISA) Secure Coding 2021	全部

OWASP ASVS	全部
OWASP Top 10 2021	全部
SANS top 25	全部
ASA Mobile Premium	全部
ASA Premium	全部
ASD STIG 5.2	全部
Top Tier	全部

排除在外:

未分類	無
Custom	無
PCI DSS v3.2.1	無
OWASP Top 10 2013	無
FISMA 2014	無
NIST SP 800-53	無
OWASP Top 10 2017	無
OWASP Mobile Top 10 2016	無
OWASP Top 10 API	無
ASD STIG 4.10	無
OWASP Top 10 2010	無
CWE top 25	無
MOIS(KISA) Secure Coding 2021	無
OWASP ASVS	無
OWASP Top 10 2021	無
SANS top 25	無
ASA Mobile Premium	無
ASA Premium	無
ASD STIG 5.2	無
Top Tier	無

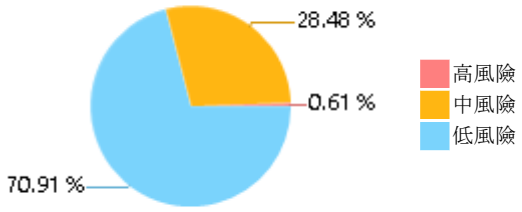
結果限制

每次問詢的結果限制設置為 50

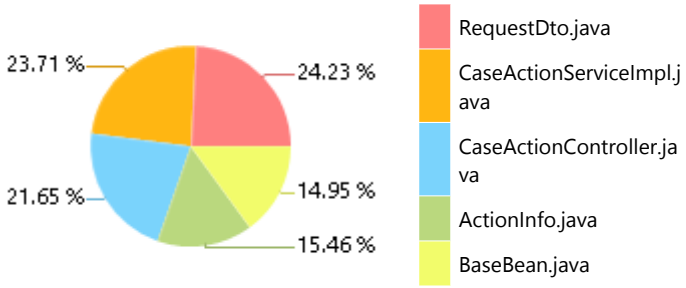
選中的問詢

選中的問詢列出在 掃描結果摘要

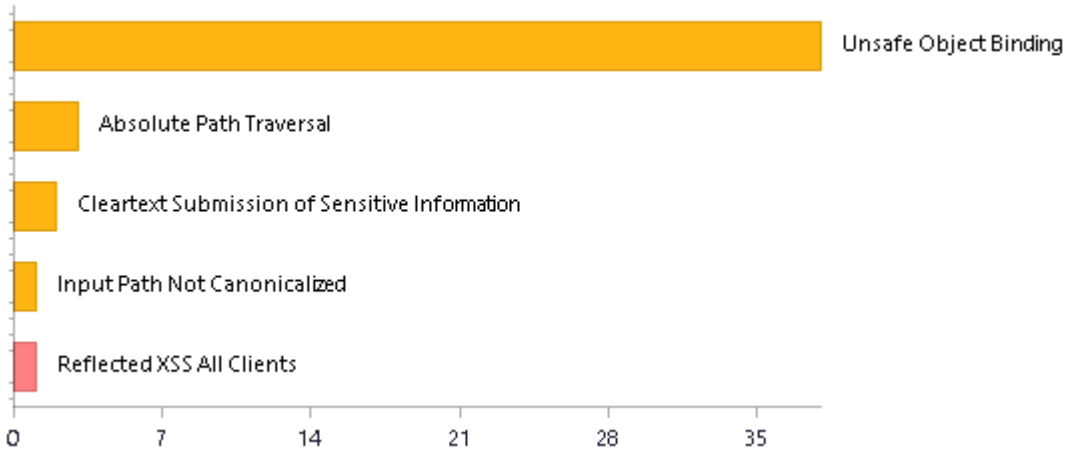
掃描結果摘要



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	8	2
A2-Broken Authentication*	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A3-Sensitive Data Exposure*	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	2	1
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	6	5
A6-Security Misconfiguration *	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	4	4
A7-Cross-Site Scripting (XSS)*	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	1	1
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	0	0
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 2021

Category	Issues Found	Best Fix Locations
A1-Broken Access Control*	18	18
A2-Cryptographic Failures	2	1
A3-Injection*	5	4
A4-Insecure Design*	33	33
A5-Security Misconfiguration*	0	0
A6-Vulnerable and Outdated Components*	0	0
A7-Identification and Authentication Failures*	59	59
A8-Software and Data Integrity Failures*	39	3
A9-Security Logging and Monitoring Failures	9	3
A10-Server-Side Request Forgery	1	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 - 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	0	0
A3-Cross-Site Scripting (XSS)*	EXTERNAL, INTERNAL, ADMIN USERS	AVERAGE	VERY WIDESPREAD	EASY	MODERATE	AFFECTED DATA AND SYSTEM	1	1
A4-Insecure Direct Object References	SYSTEM USERS	EASY	COMMON	EASY	MODERATE	EXPOSED DATA	4	3
A5-Security Misconfiguration *	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	EASY	MODERATE	ALL DATA AND SYSTEM	1	1
A6-Sensitive Data Exposure*	EXTERNAL, INTERNAL, ADMIN USERS, USERS BROWSERS	DIFFICULT	UNCOMMON	AVERAGE	SEVERE	EXPOSED DATA	2	1
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	0	0
A9-Using Components with Known Vulnerabilities*	EXTERNAL USERS, AUTOMATED TOOLS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	0	0
A10-Unvalidated Redirects and Forwards	USERS BROWSERS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	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 - PCI DSS v3.2.1

Category	Issues Found	Best Fix Locations
PCI DSS (3.2.1) - 6.5.1 - Injection flaws - particularly SQL injection	8	2
PCI DSS (3.2.1) - 6.5.2 - Buffer overflows	0	0
PCI DSS (3.2.1) - 6.5.3 - Insecure cryptographic storage*	0	0
PCI DSS (3.2.1) - 6.5.4 - Insecure communications*	2	1
PCI DSS (3.2.1) - 6.5.5 - Improper error handling*	27	27
PCI DSS (3.2.1) - 6.5.7 - Cross-site scripting (XSS)*	1	1
PCI DSS (3.2.1) - 6.5.8 - Improper access control*	4	3
PCI DSS (3.2.1) - 6.5.9 - Cross-site request forgery	0	0
PCI DSS (3.2.1) - 6.5.10 - Broken authentication and session management	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 - 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.	3	3
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.	3	2
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.	18	18
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.	0	0
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.	0	0
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.	12	6

***** 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)*	22	22
AC-4 Information Flow Enforcement (P1)	0	0
AC-6 Least Privilege (P1)	0	0
AU-9 Protection of Audit Information (P1)	8	2
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)	0	0
SC-17 Public Key Infrastructure Certificates (P1)	0	0
SC-18 Mobile Code (P2)	0	0
SC-23 Session Authenticity (P1)*	0	0
SC-28 Protection of Information at Rest (P1)*	0	0
SC-4 Information in Shared Resources (P1)	0	0
SC-5 Denial of Service Protection (P1)*	27	27
SC-8 Transmission Confidentiality and Integrity (P1)	2	1
SI-10 Information Input Validation (P1)*	2	2
SI-11 Error Handling (P2)*	1	1
SI-15 Information Output Filtering (P0)	1	1
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 - OWASP Mobile Top 10 2016

Category	Description	Issues Found	Best Fix Locations
M1-Improper Platform Usage*	This category covers misuse of a platform feature or failure to use platform security controls. It might include Android intents, platform permissions, misuse of TouchID, the Keychain, or some other security control that is part of the mobile operating system. There are several ways that mobile apps can experience this risk.	0	0
M2-Insecure Data Storage*	This category covers insecure data storage and unintended data leakage.	0	0
M3-Insecure Communication*	This category covers poor handshaking, incorrect SSL versions, weak negotiation, cleartext communication of sensitive assets, etc.	0	0
M4-Insecure Authentication*	This category captures notions of authenticating the end user or bad session management. This can include: -Failing to identify the user at all when that should be required -Failure to maintain the user's identity when it is required -Weaknesses in session management	0	0
M5-Insufficient Cryptography*	The code applies cryptography to a sensitive information asset. However, the cryptography is insufficient in some way. Note that anything and everything related to TLS or SSL goes in M3. Also, if the app fails to use cryptography at all when it should, that probably belongs in M2. This category is for issues where cryptography was attempted, but it wasnt done correctly.	0	0
M6-Insecure Authorization*	This is a category to capture any failures in authorization (e.g., authorization decisions in the client side, forced browsing, etc.). It is distinct from authentication issues (e.g., device enrolment, user identification, etc.). If the app does not authenticate users at all in a situation where it should (e.g., granting anonymous access to some resource or service when authenticated and authorized access is required), then that is an authentication failure not an authorization failure.	0	0
M7-Client Code Quality*	This category is the catch-all for code-level implementation problems in the mobile client. That's distinct from server-side coding mistakes. This would capture things like buffer overflows, format string vulnerabilities, and various other code-level mistakes where the solution is to rewrite some code that's running on the mobile device.	9	3
M8-Code Tampering*	This category covers binary patching, local resource modification, method hooking, method swizzling, and dynamic memory modification. Once the application is delivered to the mobile device, the code and data resources are resident there. An attacker can either directly modify the code, change the contents of memory dynamically, change or replace the system APIs that the application uses, or	0	0

	modify the application's data and resources. This can provide the attacker a direct method of subverting the intended use of the software for personal or monetary gain.		
M9-Reverse Engineering*	This category includes analysis of the final core binary to determine its source code, libraries, algorithms, and other assets. Software such as IDA Pro, Hopper, otool, and other binary inspection tools give the attacker insight into the inner workings of the application. This may be used to exploit other nascent vulnerabilities in the application, as well as revealing information about back end servers, cryptographic constants and ciphers, and intellectual property.	0	0
M10-Extraneous Functionality*	Often, developers include hidden backdoor functionality or other internal development security controls that are not intended to be released into a production environment. For example, a developer may accidentally include a password as a comment in a hybrid app. Another example includes disabling of 2-factor authentication during testing.	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 API

Category	Issues Found	Best Fix Locations
API1-Broken Object Level Authorization	0	0
API2-Broken Authentication*	0	0
API3-Excessive Data Exposure	1	1
API4-Lack of Resources and Rate Limiting	0	0
API5-Broken Function Level Authorization	0	0
API6-Mass Assignment	0	0
API7-Security Misconfiguration*	2	1
API8-Injection*	0	0
API9-Improper Assets Management	0	0
API10-Insufficient Logging and Monitoring	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

掃描總結 - ASD STIG 4.10

Category	Issues Found	Best Fix Locations
APSC-DV-000640 - CAT II The application must provide audit record generation capability for the renewal of session IDs.	0	0
APSC-DV-000650 - CAT II The application must not write sensitive data into the application logs.	0	0
APSC-DV-000660 - CAT II The application must provide audit record generation capability for session timeouts.	0	0
APSC-DV-000670 - CAT II The application must record a time stamp indicating when the event occurred.	0	0
APSC-DV-000680 - CAT II The application must provide audit record generation capability for HTTP headers including User-Agent, Referer, GET, and POST.	0	0
APSC-DV-000690 - CAT II The application must provide audit record generation capability for connecting system IP addresses.	0	0
APSC-DV-000700 - CAT II The application must record the username or user ID of the user associated with the event.	0	0
APSC-DV-000710 - CAT II The application must generate audit records when successful/unsuccessful attempts to grant privileges occur.	0	0
APSC-DV-000720 - CAT II The application must generate audit records when successful/unsuccessful attempts to access security objects occur.	0	0
APSC-DV-000730 - CAT II The application must generate audit records when successful/unsuccessful attempts to access security levels occur.	0	0
APSC-DV-000740 - CAT II The application must generate audit records when successful/unsuccessful attempts to access categories of information (e.g., classification levels) occur.	0	0
APSC-DV-000750 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify privileges occur.	0	0
APSC-DV-000760 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify security objects occur.	0	0
APSC-DV-000770 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify security levels occur.	0	0
APSC-DV-000780 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify categories of information (e.g., classification levels) occur.	0	0
APSC-DV-000790 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete privileges occur.	0	0
APSC-DV-000800 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete security levels occur.	0	0
APSC-DV-000810 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete application database security objects occur.	0	0
APSC-DV-000820 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete categories of information (e.g., classification levels) occur.	0	0
APSC-DV-000830 - CAT II The application must generate audit records when successful/unsuccessful logon attempts occur.	0	0
APSC-DV-000840 - CAT II The application must generate audit records for privileged activities or other system-level access.	0	0
APSC-DV-000850 - CAT II The application must generate audit records showing starting and ending time for user access to the system.	0	0
APSC-DV-000860 - CAT II The application must generate audit records when successful/unsuccessful accesses to objects occur.	0	0

APSC-DV-000870 - CAT II The application must generate audit records for all direct access to the information system.	0	0
APSC-DV-000880 - CAT II The application must generate audit records for all account creations, modifications, disabling, and termination events.	0	0
APSC-DV-000910 - CAT II The application must initiate session auditing upon startup.	0	0
APSC-DV-000940 - CAT II The application must log application shutdown events.	0	0
APSC-DV-000950 - CAT II The application must log destination IP addresses.	0	0
APSC-DV-000960 - CAT II The application must log user actions involving access to data.	0	0
APSC-DV-000970 - CAT II The application must log user actions involving changes to data.	0	0
APSC-DV-000980 - CAT II The application must produce audit records containing information to establish when (date and time) the events occurred.	0	0
APSC-DV-000990 - CAT II The application must produce audit records containing enough information to establish which component, feature or function of the application triggered the audit event.	0	0
APSC-DV-001000 - CAT II When using centralized logging; the application must include a unique identifier in order to distinguish itself from other application logs.	0	0
APSC-DV-001010 - CAT II The application must produce audit records that contain information to establish the outcome of the events.	0	0
APSC-DV-001020 - CAT II The application must generate audit records containing information that establishes the identity of any individual or process associated with the event.	0	0
APSC-DV-001030 - CAT II The application must generate audit records containing the full-text recording of privileged commands or the individual identities of group account users.	0	0
APSC-DV-001040 - CAT II The application must implement transaction recovery logs when transaction based.	0	0
APSC-DV-001050 - CAT II The application must provide centralized management and configuration of the content to be captured in audit records generated by all application components.	0	0
APSC-DV-001070 - CAT II The application must off-load audit records onto a different system or media than the system being audited.	0	0
APSC-DV-001080 - CAT II The application must be configured to write application logs to a centralized log repository.	0	0
APSC-DV-001090 - CAT II The application must provide an immediate warning to the SA and ISSO (at a minimum) when allocated audit record storage volume reaches 75% of repository maximum audit record storage capacity.	0	0
APSC-DV-001100 - CAT II Applications categorized as having a moderate or high impact must provide an immediate real-time alert to the SA and ISSO (at a minimum) for all audit failure events.	0	0
APSC-DV-001110 - CAT II The application must alert the ISSO and SA (at a minimum) in the event of an audit processing failure.	0	0
APSC-DV-001120 - CAT II The application must shut down by default upon audit failure (unless availability is an overriding concern).	0	0
APSC-DV-001130 - CAT II The application must provide the capability to centrally review and analyze audit records from multiple components within the system.	0	0
APSC-DV-001140 - CAT II The application must provide the capability to filter audit records for events of interest based upon organization-defined criteria.	0	0
APSC-DV-001150 - CAT II The application must provide an audit reduction capability that supports on-demand reporting requirements.	0	0
APSC-DV-001160 - CAT II The application must provide an audit reduction capability that supports on-demand audit review and analysis.	0	0
APSC-DV-001170 - CAT II The application must provide an audit reduction capability that supports after-the-fact investigations of security incidents.	0	0
APSC-DV-001180 - CAT II The application must provide a report generation capability that supports on-demand audit review and analysis.	0	0

APSC-DV-001190 - CAT II The application must provide a report generation capability that supports on-demand reporting requirements.	0	0
APSC-DV-001200 - CAT II The application must provide a report generation capability that supports after-the-fact investigations of security incidents.	0	0
APSC-DV-001210 - CAT II The application must provide an audit reduction capability that does not alter original content or time ordering of audit records.	0	0
APSC-DV-001220 - CAT II The application must provide a report generation capability that does not alter original content or time ordering of audit records.	0	0
APSC-DV-001250 - CAT II The applications must use internal system clocks to generate time stamps for audit records.	0	0
APSC-DV-001260 - CAT II The application must record time stamps for audit records that can be mapped to Coordinated Universal Time (UTC) or Greenwich Mean Time (GMT).	0	0
APSC-DV-001270 - CAT II The application must record time stamps for audit records that meet a granularity of one second for a minimum degree of precision.	0	0
APSC-DV-001280 - CAT II The application must protect audit information from any type of unauthorized read access.	0	0
APSC-DV-001290 - CAT II The application must protect audit information from unauthorized modification.	0	0
APSC-DV-001300 - CAT II The application must protect audit information from unauthorized deletion.	0	0
APSC-DV-001310 - CAT II The application must protect audit tools from unauthorized access.	0	0
APSC-DV-001320 - CAT II The application must protect audit tools from unauthorized modification.	0	0
APSC-DV-001330 - CAT II The application must protect audit tools from unauthorized deletion.	0	0
APSC-DV-001340 - CAT II The application must back up audit records at least every seven days onto a different system or system component than the system or component being audited.	0	0
APSC-DV-001570 - CAT II The application must electronically verify Personal Identity Verification (PIV) credentials.	0	0
APSC-DV-001350 - CAT II The application must use cryptographic mechanisms to protect the integrity of audit information.	0	0
APSC-DV-001360 - CAT II Application audit tools must be cryptographically hashed.	0	0
APSC-DV-001370 - CAT II The integrity of the audit tools must be validated by checking the files for changes in the cryptographic hash value.	0	0
APSC-DV-001390 - CAT II The application must prohibit user installation of software without explicit privileged status.	0	0
APSC-DV-001410 - CAT II The application must enforce access restrictions associated with changes to application configuration.	0	0
APSC-DV-001420 - CAT II The application must audit who makes configuration changes to the application.	0	0
APSC-DV-001430 - CAT II The application must have the capability to prevent the installation of patches, service packs, or application components without verification the software component has been digitally signed using a certificate that is recognized and approved by the orga	0	0
APSC-DV-001440 - CAT II The applications must limit privileges to change the software resident within software libraries.	0	0
APSC-DV-001460 - CAT II An application vulnerability assessment must be conducted.	0	0
APSC-DV-001480 - CAT II The application must prevent program execution in accordance with organization-defined policies regarding software program usage and restrictions, and/or rules authorizing the terms and conditions of software program usage.	0	0
APSC-DV-001490 - CAT II The application must employ a deny-all, permit-by-exception (whitelist) policy to allow the execution of authorized software programs.	0	0
APSC-DV-001500 - CAT II The application must be configured to disable non-essential capabilities.	0	0
APSC-DV-001510 - CAT II The application must be configured to use only functions, ports, and protocols permitted to it in the PPSM CAL.	0	0

APSC-DV-001520 - CAT II The application must require users to reauthenticate when organization-defined circumstances or situations require reauthentication.	0	0
APSC-DV-001530 - CAT II The application must require devices to reauthenticate when organization-defined circumstances or situations requiring reauthentication.	0	0
APSC-DV-001540 - CAT I The application must uniquely identify and authenticate organizational users (or processes acting on behalf of organizational users).	0	0
APSC-DV-001550 - CAT II The application must use multifactor (Alt. Token) authentication for network access to privileged accounts.	0	0
APSC-DV-001560 - CAT II The application must accept Personal Identity Verification (PIV) credentials.	0	0
APSC-DV-001580 - CAT II The application must use multifactor (e.g., CAC, Alt. Token) authentication for network access to non-privileged accounts.	0	0
APSC-DV-001590 - CAT II The application must use multifactor (Alt. Token) authentication for local access to privileged accounts.	0	0
APSC-DV-001600 - CAT II The application must use multifactor (e.g., CAC, Alt. Token) authentication for local access to non-privileged accounts.	0	0
APSC-DV-001610 - CAT II The application must ensure users are authenticated with an individual authenticator prior to using a group authenticator.	0	0
APSC-DV-001620 - CAT II The application must implement replay-resistant authentication mechanisms for network access to privileged accounts.	0	0
APSC-DV-001630 - CAT II The application must implement replay-resistant authentication mechanisms for network access to non-privileged accounts.	0	0
APSC-DV-001640 - CAT II The application must utilize mutual authentication when endpoint device non-repudiation protections are required by DoD policy or by the data owner.	0	0
APSC-DV-001650 - CAT II The application must authenticate all network connected endpoint devices before establishing any connection.	0	0
APSC-DV-001660 - CAT II Service-Oriented Applications handling non-releasable data must authenticate endpoint devices via mutual SSL/TLS.	0	0
APSC-DV-001670 - CAT II The application must disable device identifiers after 35 days of inactivity unless a cryptographic certificate is used for authentication.	0	0
APSC-DV-001680 - CAT I The application must enforce a minimum 15-character password length.	0	0
APSC-DV-001690 - CAT II The application must enforce password complexity by requiring that at least one upper-case character be used.	0	0
APSC-DV-001700 - CAT II The application must enforce password complexity by requiring that at least one lower-case character be used.	0	0
APSC-DV-001710 - CAT II The application must enforce password complexity by requiring that at least one numeric character be used.	0	0
APSC-DV-001720 - CAT II The application must enforce password complexity by requiring that at least one special character be used.	0	0
APSC-DV-001730 - CAT II The application must require the change of at least 8 of the total number of characters when passwords are changed.	0	0
APSC-DV-001740 - CAT I The application must only store cryptographic representations of passwords.	0	0
APSC-DV-001850 - CAT I The application must not display passwords/PINs as clear text.	0	0
APSC-DV-001750 - CAT I The application must transmit only cryptographically-protected passwords.	0	0
APSC-DV-001760 - CAT II The application must enforce 24 hours/1 day as the minimum password lifetime.	0	0
APSC-DV-001770 - CAT II The application must enforce a 60-day maximum password lifetime restriction.	0	0
APSC-DV-001780 - CAT II The application must prohibit password reuse for a minimum of five generations.	0	0
APSC-DV-001790 - CAT II The application must allow the use of a temporary password for system logons with an immediate change to a permanent password.	0	0

APSC-DV-001795 - CAT II The application password must not be changeable by users other than the administrator or the user with which the password is associated.	0	0
APSC-DV-001800 - CAT II The application must terminate existing user sessions upon account deletion.	0	0
APSC-DV-001820 - CAT I The application, when using PKI-based authentication, must enforce authorized access to the corresponding private key.	0	0
APSC-DV-001830 - CAT II The application must map the authenticated identity to the individual user or group account for PKI-based authentication.	0	0
APSC-DV-001870 - CAT II The application must uniquely identify and authenticate non-organizational users (or processes acting on behalf of non-organizational users).	0	0
APSC-DV-001810 - CAT I The application, when utilizing PKI-based authentication, must validate certificates by constructing a certification path (which includes status information) to an accepted trust anchor.	0	0
APSC-DV-001840 - CAT II The application, for PKI-based authentication, must implement a local cache of revocation data to support path discovery and validation in case of the inability to access revocation information via the network.	0	0
APSC-DV-001860 - CAT II The application must use mechanisms meeting the requirements of applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidance for authentication to a cryptographic module.	0	0
APSC-DV-001880 - CAT II The application must accept Personal Identity Verification (PIV) credentials from other federal agencies.	0	0
APSC-DV-001890 - CAT II The application must electronically verify Personal Identity Verification (PIV) credentials from other federal agencies.	0	0
APSC-DV-002050 - CAT II Applications making SAML assertions must use FIPS-approved random numbers in the generation of SessionIndex in the SAML element AuthnStatement.	0	0
APSC-DV-001900 - CAT II The application must accept FICAM-approved third-party credentials.	0	0
APSC-DV-001910 - CAT II The application must conform to FICAM-issued profiles.	0	0
APSC-DV-001930 - CAT II Applications used for non-local maintenance sessions must audit non-local maintenance and diagnostic sessions for organization-defined auditable events.	0	0
APSC-DV-000310 - CAT III The application must have a process, feature or function that prevents removal or disabling of emergency accounts.	0	0
APSC-DV-001940 - CAT II Applications used for non-local maintenance sessions must implement cryptographic mechanisms to protect the integrity of non-local maintenance and diagnostic communications.	0	0
APSC-DV-001950 - CAT II Applications used for non-local maintenance sessions must implement cryptographic mechanisms to protect the confidentiality of non-local maintenance and diagnostic communications.	0	0
APSC-DV-001960 - CAT II Applications used for non-local maintenance sessions must verify remote disconnection at the termination of non-local maintenance and diagnostic sessions.	0	0
APSC-DV-001970 - CAT II The application must employ strong authenticators in the establishment of non-local maintenance and diagnostic sessions.	0	0
APSC-DV-001980 - CAT II The application must terminate all sessions and network connections when non-local maintenance is completed.	0	0
APSC-DV-001995 - CAT II The application must not be vulnerable to race conditions.	0	0
APSC-DV-002000 - CAT II The application must terminate all network connections associated with a communications session at the end of the session.	0	0
APSC-DV-002010 - CAT II The application must implement NSA-approved cryptography to protect classified information in accordance with applicable federal laws, Executive Orders, directives, policies, regulations, and standards.	0	0
APSC-DV-002020 - CAT II The application must utilize FIPS-validated cryptographic modules when signing application components.	0	0
APSC-DV-002030 - CAT II The application must utilize FIPS-validated cryptographic modules when generating cryptographic hashes.	0	0

APSC-DV-002040 - CAT II The application must utilize FIPS-validated cryptographic modules when protecting unclassified information that requires cryptographic protection.	0	0
APSC-DV-002150 - CAT II The application user interface must be either physically or logically separated from data storage and management interfaces.	0	0
APSC-DV-002210 - CAT II The application must set the HTTPOnly flag on session cookies.	0	0
APSC-DV-002220 - CAT II The application must set the secure flag on session cookies.	0	0
APSC-DV-002230 - CAT I The application must not expose session IDs.	0	0
APSC-DV-002240 - CAT I The application must destroy the session ID value and/or cookie on logoff or browser close.	0	0
APSC-DV-002250 - CAT II Applications must use system-generated session identifiers that protect against session fixation.	0	0
APSC-DV-002260 - CAT II Applications must validate session identifiers.	0	0
APSC-DV-002270 - CAT II Applications must not use URL embedded session IDs.	0	0
APSC-DV-002280 - CAT II The application must not re-use or recycle session IDs.	0	0
APSC-DV-002290 - CAT II The application must use the Federal Information Processing Standard (FIPS) 140-2-validated cryptographic modules and random number generator if the application implements encryption, key exchange, digital signature, and hash functionality.	0	0
APSC-DV-002300 - CAT II The application must only allow the use of DoD-approved certificate authorities for verification of the establishment of protected sessions.	0	0
APSC-DV-002310 - CAT I The application must fail to a secure state if system initialization fails, shutdown fails, or aborts fail.	0	0
APSC-DV-002320 - CAT II In the event of a system failure, applications must preserve any information necessary to determine cause of failure and any information necessary to return to operations with least disruption to mission processes.	0	0
APSC-DV-002330 - CAT II The application must protect the confidentiality and integrity of stored information when required by DoD policy or the information owner.	0	0
APSC-DV-002340 - CAT II The application must implement approved cryptographic mechanisms to prevent unauthorized modification of organization-defined information at rest on organization-defined information system components.	0	0
APSC-DV-002350 - CAT II The application must use appropriate cryptography in order to protect stored DoD information when required by the information owner or DoD policy.	0	0
APSC-DV-002360 - CAT II The application must isolate security functions from non-security functions.	0	0
APSC-DV-002370 - CAT II The application must maintain a separate execution domain for each executing process.	0	0
APSC-DV-002380 - CAT II Applications must prevent unauthorized and unintended information transfer via shared system resources.	0	0
APSC-DV-002390 - CAT II XML-based applications must mitigate DoS attacks by using XML filters, parser options, or gateways.	0	0
APSC-DV-002400 - CAT II The application must restrict the ability to launch Denial of Service (DoS) attacks against itself or other information systems.	0	0
APSC-DV-002410 - CAT II The web service design must include redundancy mechanisms when used with high-availability systems.	0	0
APSC-DV-002420 - CAT II An XML firewall function must be deployed to protect web services when exposed to untrusted networks.	0	0
APSC-DV-002610 - CAT II The application must remove organization-defined software components after updated versions have been installed.	0	0
APSC-DV-002440 - CAT I The application must protect the confidentiality and integrity of transmitted information.	0	0
APSC-DV-002450 - CAT II The application must implement cryptographic mechanisms to prevent unauthorized disclosure of information and/or detect changes to information during transmission unless otherwise protected by alternative physical safeguards, such as, at a minimum, a Prot	0	0

APSC-DV-002460 - CAT II The application must maintain the confidentiality and integrity of information during preparation for transmission.	0	0
APSC-DV-002470 - CAT II The application must maintain the confidentiality and integrity of information during reception.	0	0
APSC-DV-002480 - CAT II The application must not disclose unnecessary information to users.	0	0
APSC-DV-002485 - CAT I The application must not store sensitive information in hidden fields.	0	0
APSC-DV-002490 - CAT I The application must protect from Cross-Site Scripting (XSS) vulnerabilities.	0	0
APSC-DV-002500 - CAT II The application must protect from Cross-Site Request Forgery (CSRF) vulnerabilities.	0	0
APSC-DV-002510 - CAT I The application must protect from command injection.	0	0
APSC-DV-002520 - CAT II The application must protect from canonical representation vulnerabilities.	0	0
APSC-DV-002530 - CAT II The application must validate all input.	0	0
APSC-DV-002540 - CAT I The application must not be vulnerable to SQL Injection.	0	0
APSC-DV-002550 - CAT I The application must not be vulnerable to XML-oriented attacks.	0	0
APSC-DV-002560 - CAT I The application must not be subject to input handling vulnerabilities.	0	0
APSC-DV-002570 - CAT II The application must generate error messages that provide information necessary for corrective actions without revealing information that could be exploited by adversaries.	0	0
APSC-DV-002580 - CAT II The application must reveal error messages only to the ISSO, ISSM, or SA.	0	0
APSC-DV-002590 - CAT I The application must not be vulnerable to overflow attacks.	0	0
APSC-DV-002630 - CAT II Security-relevant software updates and patches must be kept up to date.	0	0
APSC-DV-002760 - CAT II The application performing organization-defined security functions must verify correct operation of security functions.	0	0
APSC-DV-002900 - CAT II The ISSO must ensure application audit trails are retained for at least 1 year for applications without SAMI data, and 5 years for applications including SAMI data.	0	0
APSC-DV-002770 - CAT II The application must perform verification of the correct operation of security functions: upon system startup and/or restart; upon command by a user with privileged access; and/or every 30 days.	0	0
APSC-DV-002780 - CAT III The application must notify the ISSO and ISSM of failed security verification tests.	0	0
APSC-DV-002870 - CAT II Unsigned Category 1A mobile code must not be used in the application in accordance with DoD policy.	0	0
APSC-DV-002880 - CAT II The ISSO must ensure an account management process is implemented, verifying only authorized users can gain access to the application, and individual accounts designated as inactive, suspended, or terminated are promptly removed.	0	0
APSC-DV-002890 - CAT I Application web servers must be on a separate network segment from the application and database servers if it is a tiered application operating in the DoD DMZ.	0	0
APSC-DV-002910 - CAT II The ISSO must review audit trails periodically based on system documentation recommendations or immediately upon system security events.	0	0
APSC-DV-002920 - CAT II The ISSO must report all suspected violations of IA policies in accordance with DoD information system IA procedures.	0	0
APSC-DV-002930 - CAT II The ISSO must ensure active vulnerability testing is performed.	0	0
APSC-DV-002980 - CAT II New IP addresses, data services, and associated ports used by the application must be submitted to the appropriate approving authority for the organization, which in turn will be submitted through the DoD Ports, Protocols, and Services Management (DoD PPS	0	0
APSC-DV-002950 - CAT II Execution flow diagrams and design documents must be created to show how deadlock and recursion issues in web services are being mitigated.	0	0
APSC-DV-002960 - CAT II The designer must ensure the application does not store configuration and control files in the same directory as user data.	0	0
APSC-DV-002970 - CAT II The ISSO must ensure if a DoD STIG or NSA guide is not available, a third-party	0	0

product will be configured by following available guidance.		
APSC-DV-002990 - CAT II The application must be registered with the DoD Ports and Protocols Database.	0	0
APSC-DV-002990 - CAT II The application must be registered with the DoD Ports and Protocols Database.	0	0
APSC-DV-002995 - CAT II The Configuration Management (CM) repository must be properly patched and STIG compliant.	0	0
APSC-DV-003000 - CAT II Access privileges to the Configuration Management (CM) repository must be reviewed every three months.	0	0
APSC-DV-003010 - CAT II A Software Configuration Management (SCM) plan describing the configuration control and change management process of application objects developed by the organization and the roles and responsibilities of the organization must be created and maintained.	0	0
APSC-DV-003020 - CAT II A Configuration Control Board (CCB) that meets at least every release cycle, for managing the Configuration Management (CM) process must be established.	0	0
APSC-DV-003030 - CAT II The application services and interfaces must be compatible with and ready for IPv6 networks.	0	0
APSC-DV-003040 - CAT II The application must not be hosted on a general purpose machine if the application is designated as critical or high availability by the ISSO.	0	0
APSC-DV-003050 - CAT II A disaster recovery/continuity plan must exist in accordance with DoD policy based on the applications availability requirements.	0	0
APSC-DV-003060 - CAT II Recovery procedures and technical system features must exist so recovery is performed in a secure and verifiable manner. The ISSO will document circumstances inhibiting a trusted recovery.	0	0
APSC-DV-003070 - CAT II Data backup must be performed at required intervals in accordance with DoD policy.	0	0
APSC-DV-003080 - CAT II Back-up copies of the application software or source code must be stored in a fire-rated container or stored separately (offsite).	0	0
APSC-DV-003090 - CAT II Procedures must be in place to assure the appropriate physical and technical protection of the backup and restoration of the application.	0	0
APSC-DV-003100 - CAT II The application must use encryption to implement key exchange and authenticate endpoints prior to establishing a communication channel for key exchange.	0	0
APSC-DV-003110 - CAT I The application must not contain embedded authentication data.	0	0
APSC-DV-003120 - CAT I The application must have the capability to mark sensitive/classified output when required.	0	0
APSC-DV-003130 - CAT III Prior to each release of the application, updates to system, or applying patches; tests plans and procedures must be created and executed.	0	0
APSC-DV-003150 - CAT II At least one tester must be designated to test for security flaws in addition to functional testing.	0	0
APSC-DV-003140 - CAT II Application files must be cryptographically hashed prior to deploying to DoD operational networks.	0	0
APSC-DV-003160 - CAT III Test procedures must be created and at least annually executed to ensure system initialization, shutdown, and aborts are configured to verify the system remains in a secure state.	0	0
APSC-DV-003170 - CAT II An application code review must be performed on the application.	0	0
APSC-DV-003180 - CAT III Code coverage statistics must be maintained for each release of the application.	0	0
APSC-DV-003190 - CAT II Flaws found during a code review must be tracked in a defect tracking system.	0	0
APSC-DV-003200 - CAT II The changes to the application must be assessed for IA and accreditation impact prior to implementation.	0	0
APSC-DV-003210 - CAT II Security flaws must be fixed or addressed in the project plan.	0	0
APSC-DV-003215 - CAT III The application development team must follow a set of coding standards.	0	0
APSC-DV-003220 - CAT III The designer must create and update the Design Document for each release of the application.	0	0

APSC-DV-003230 - CAT II Threat models must be documented and reviewed for each application release and updated as required by design and functionality changes or when new threats are discovered.	0	0
APSC-DV-003235 - CAT II The application must not be subject to error handling vulnerabilities.	0	0
APSC-DV-003250 - CAT I The application must be decommissioned when maintenance or support is no longer available.	0	0
APSC-DV-003236 - CAT II The application development team must provide an application incident response plan.	0	0
APSC-DV-003240 - CAT I All products must be supported by the vendor or the development team.	0	0
APSC-DV-003260 - CAT III Procedures must be in place to notify users when an application is decommissioned.	0	0
APSC-DV-003270 - CAT II Unnecessary built-in application accounts must be disabled.	0	0
APSC-DV-003280 - CAT I Default passwords must be changed.	0	0
APSC-DV-003330 - CAT II The system must alert an administrator when low resource conditions are encountered.	0	0
APSC-DV-003285 - CAT II An Application Configuration Guide must be created and included with the application.	0	0
APSC-DV-003290 - CAT II If the application contains classified data, a Security Classification Guide must exist containing data elements and their classification.	0	0
APSC-DV-003300 - CAT II The designer must ensure uncategorized or emerging mobile code is not used in applications.	0	0
APSC-DV-003310 - CAT II Production database exports must have database administration credentials and sensitive data removed before releasing the export.	0	0
APSC-DV-003320 - CAT II Protections against DoS attacks must be implemented.	0	0
APSC-DV-003340 - CAT III At least one application administrator must be registered to receive update notifications, or security alerts, when automated alerts are available.	0	0
APSC-DV-003360 - CAT III The application must generate audit records when concurrent logons from different workstations occur.	0	0
APSC-DV-003345 - CAT III The application must provide notifications or alerts when product update and security related patches are available.	0	0
APSC-DV-003350 - CAT II Connections between the DoD enclave and the Internet or other public or commercial wide area networks must require a DMZ.	0	0
APSC-DV-003400 - CAT II The Program Manager must verify all levels of program management, designers, developers, and testers receive annual security training pertaining to their job function.	0	0
APSC-DV-000010 - CAT II The application must provide a capability to limit the number of logon sessions per user.	0	0
APSC-DV-000060 - CAT II The application must clear temporary storage and cookies when the session is terminated.	0	0
APSC-DV-000070 - CAT II The application must automatically terminate the non-privileged user session and log off non-privileged users after a 15 minute idle time period has elapsed.	0	0
APSC-DV-000080 - CAT II The application must automatically terminate the admin user session and log off admin users after a 10 minute idle time period is exceeded.	0	0
APSC-DV-000090 - CAT II Applications requiring user access authentication must provide a logoff capability for user initiated communication session.	0	0
APSC-DV-000100 - CAT III The application must display an explicit logoff message to users indicating the reliable termination of authenticated communications sessions.	0	0
APSC-DV-000110 - CAT II The application must associate organization-defined types of security attributes having organization-defined security attribute values with information in storage.	0	0
APSC-DV-000120 - CAT II The application must associate organization-defined types of security attributes having organization-defined security attribute values with information in process.	0	0
APSC-DV-000130 - CAT II The application must associate organization-defined types of security attributes	0	0

having organization-defined security attribute values with information in transmission.		
APSC-DV-000160 - CAT II The application must implement DoD-approved encryption to protect the confidentiality of remote access sessions.	0	0
APSC-DV-000170 - CAT II The application must implement cryptographic mechanisms to protect the integrity of remote access sessions.	0	0
APSC-DV-000190 - CAT I Messages protected with WS_Security must use time stamps with creation and expiration times.	0	0
APSC-DV-000180 - CAT II Applications with SOAP messages requiring integrity must include the following message elements:-Message ID-Service Request-Timestamp-SAML Assertion (optionally included in messages) and all elements of the message must be digitally signed.	0	0
APSC-DV-000200 - CAT I Validity periods must be verified on all application messages using WS-Security or SAML assertions.	0	0
APSC-DV-000210 - CAT II The application must ensure each unique asserting party provides unique assertion ID references for each SAML assertion.	0	0
APSC-DV-000220 - CAT II The application must ensure encrypted assertions, or equivalent confidentiality protections are used when assertion data is passed through an intermediary, and confidentiality of the assertion data is required when passing through the intermediary.	0	0
APSC-DV-000230 - CAT I The application must use the NotOnOrAfter condition when using the SubjectConfirmation element in a SAML assertion.	0	0
APSC-DV-000240 - CAT I The application must use both the NotBefore and NotOnOrAfter elements or OneTimeUse element when using the Conditions element in a SAML assertion.	0	0
APSC-DV-000250 - CAT II The application must ensure if a OneTimeUse element is used in an assertion, there is only one of the same used in the Conditions element portion of an assertion.	0	0
APSC-DV-000260 - CAT II The application must ensure messages are encrypted when the SessionIndex is tied to privacy data.	0	0
APSC-DV-000290 - CAT II Shared/group account credentials must be terminated when members leave the group.	0	0
APSC-DV-000280 - CAT II The application must provide automated mechanisms for supporting account management functions.	0	0
APSC-DV-000300 - CAT II The application must automatically remove or disable temporary user accounts 72 hours after account creation.	0	0
APSC-DV-000320 - CAT III The application must automatically disable accounts after a 35 day period of account inactivity.	0	0
APSC-DV-000330 - CAT II Unnecessary application accounts must be disabled, or deleted.	0	0
APSC-DV-000420 - CAT II The application must automatically audit account enabling actions.	0	0
APSC-DV-000340 - CAT II The application must automatically audit account creation.	0	0
APSC-DV-000350 - CAT II The application must automatically audit account modification.	0	0
APSC-DV-000360 - CAT II The application must automatically audit account disabling actions.	0	0
APSC-DV-000370 - CAT II The application must automatically audit account removal actions.	0	0
APSC-DV-000380 - CAT III The application must notify System Administrators and Information System Security Officers when accounts are created.	0	0
APSC-DV-000390 - CAT III The application must notify System Administrators and Information System Security Officers when accounts are modified.	0	0
APSC-DV-000400 - CAT III The application must notify System Administrators and Information System Security Officers of account disabling actions.	0	0
APSC-DV-000410 - CAT III The application must notify System Administrators and Information System Security Officers of account removal actions.	0	0
APSC-DV-000430 - CAT III The application must notify System Administrators and Information System Security Officers of account enabling actions.	0	0
APSC-DV-000440 - CAT II Application data protection requirements must be identified and documented.	0	0

APSC-DV-000520 - CAT II The application must audit the execution of privileged functions.	0	0
APSC-DV-000450 - CAT II The application must utilize organization-defined data mining detection techniques for organization-defined data storage objects to adequately detect data mining attempts.	0	0
APSC-DV-000460 - CAT I The application must enforce approved authorizations for logical access to information and system resources in accordance with applicable access control policies.	0	0
APSC-DV-000470 - CAT II The application must enforce organization-defined discretionary access control policies over defined subjects and objects.	0	0
APSC-DV-000480 - CAT II The application must enforce approved authorizations for controlling the flow of information within the system based on organization-defined information flow control policies.	0	0
APSC-DV-000490 - CAT II The application must enforce approved authorizations for controlling the flow of information between interconnected systems based on organization-defined information flow control policies.	0	0
APSC-DV-000500 - CAT II The application must prevent non-privileged users from executing privileged functions to include disabling, circumventing, or altering implemented security safeguards/countermeasures.	0	0
APSC-DV-000510 - CAT I The application must execute without excessive account permissions.	0	0
APSC-DV-000530 - CAT I The application must enforce the limit of three consecutive invalid logon attempts by a user during a 15 minute time period.	0	0
APSC-DV-000560 - CAT III The application must retain the Standard Mandatory DoD Notice and Consent Banner on the screen until users acknowledge the usage conditions and take explicit actions to log on for further access.	0	0
APSC-DV-000540 - CAT II The application administrator must follow an approved process to unlock locked user accounts.	0	0
APSC-DV-000550 - CAT III The application must display the Standard Mandatory DoD Notice and Consent Banner before granting access to the application.	0	0
APSC-DV-000570 - CAT III The publicly accessible application must display the Standard Mandatory DoD Notice and Consent Banner before granting access to the application.	0	0
APSC-DV-000580 - CAT III The application must display the time and date of the users last successful logon.	0	0
APSC-DV-000630 - CAT II The application must provide audit record generation capability for the destruction of session IDs.	0	0
APSC-DV-000590 - CAT II The application must protect against an individual (or process acting on behalf of an individual) falsely denying having performed organization-defined actions to be covered by non-repudiation.	0	0
APSC-DV-000600 - CAT II For applications providing audit record aggregation, the application must compile audit records from organization-defined information system components into a system-wide audit trail that is time-correlated with an organization-defined level of tolerance	0	0
APSC-DV-000610 - CAT II The application must provide the capability for organization-identified individuals or roles to change the auditing to be performed on all application components, based on all selectable event criteria within organization-defined time thresholds.	0	0
APSC-DV-000620 - CAT II The application must provide audit record generation capability for the creation of session IDs.	0	0

掃描總結 - ASD STIG 5.2

Category	Issues Found	Best Fix Locations
APSC-DV-000640 - CAT II The application must provide audit record generation capability for the renewal of session IDs.	0	0
APSC-DV-000650 - CAT II The application must not write sensitive data into the application logs.	0	0
APSC-DV-000660 - CAT II The application must provide audit record generation capability for session timeouts.	0	0
APSC-DV-000670 - CAT II The application must record a time stamp indicating when the event occurred.	0	0
APSC-DV-000680 - CAT II The application must provide audit record generation capability for HTTP headers including User-Agent, Referer, GET, and POST.	0	0
APSC-DV-000690 - CAT II The application must provide audit record generation capability for connecting system IP addresses.	0	0
APSC-DV-000700 - CAT II The application must record the username or user ID of the user associated with the event.	0	0
APSC-DV-000710 - CAT II The application must generate audit records when successful/unsuccessful attempts to grant privileges occur.	0	0
APSC-DV-000720 - CAT II The application must generate audit records when successful/unsuccessful attempts to access security objects occur.	0	0
APSC-DV-000730 - CAT II The application must generate audit records when successful/unsuccessful attempts to access security levels occur.	0	0
APSC-DV-000740 - CAT II The application must generate audit records when successful/unsuccessful attempts to access categories of information (e.g., classification levels) occur.	0	0
APSC-DV-000750 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify privileges occur.	0	0
APSC-DV-000760 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify security objects occur.	0	0
APSC-DV-000770 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify security levels occur.	0	0
APSC-DV-000780 - CAT II The application must generate audit records when successful/unsuccessful attempts to modify categories of information (e.g., classification levels) occur.	0	0
APSC-DV-000790 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete privileges occur.	0	0
APSC-DV-000800 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete security levels occur.	0	0
APSC-DV-000810 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete application database security objects occur.	0	0
APSC-DV-000820 - CAT II The application must generate audit records when successful/unsuccessful attempts to delete categories of information (e.g., classification levels) occur.	0	0
APSC-DV-000830 - CAT II The application must generate audit records when successful/unsuccessful logon attempts occur.	0	0
APSC-DV-000840 - CAT II The application must generate audit records for privileged activities or other system-level access.	0	0
APSC-DV-000850 - CAT II The application must generate audit records showing starting and ending time for user access to the system.	0	0
APSC-DV-000860 - CAT II The application must generate audit records when successful/unsuccessful accesses to objects occur.	0	0

APSC-DV-000870 - CAT II The application must generate audit records for all direct access to the information system.	0	0
APSC-DV-000880 - CAT II The application must generate audit records for all account creations, modifications, disabling, and termination events.	0	0
APSC-DV-000910 - CAT II The application must initiate session auditing upon startup.	0	0
APSC-DV-000940 - CAT II The application must log application shutdown events.	0	0
APSC-DV-000950 - CAT II The application must log destination IP addresses.	0	0
APSC-DV-000960 - CAT II The application must log user actions involving access to data.	0	0
APSC-DV-000970 - CAT II The application must log user actions involving changes to data.	0	0
APSC-DV-000980 - CAT II The application must produce audit records containing information to establish when (date and time) the events occurred.	0	0
APSC-DV-000990 - CAT II The application must produce audit records containing enough information to establish which component, feature or function of the application triggered the audit event.	0	0
APSC-DV-001000 - CAT II When using centralized logging; the application must include a unique identifier in order to distinguish itself from other application logs.	0	0
APSC-DV-001010 - CAT II The application must produce audit records that contain information to establish the outcome of the events.	0	0
APSC-DV-001020 - CAT II The application must generate audit records containing information that establishes the identity of any individual or process associated with the event.	0	0
APSC-DV-001030 - CAT II The application must generate audit records containing the full-text recording of privileged commands or the individual identities of group account users.	0	0
APSC-DV-001040 - CAT II The application must implement transaction recovery logs when transaction based.	0	0
APSC-DV-001050 - CAT II The application must provide centralized management and configuration of the content to be captured in audit records generated by all application components.	0	0
APSC-DV-001070 - CAT II The application must off-load audit records onto a different system or media than the system being audited.	0	0
APSC-DV-001080 - CAT II The application must be configured to write application logs to a centralized log repository.	0	0
APSC-DV-001090 - CAT II The application must provide an immediate warning to the SA and ISSO (at a minimum) when allocated audit record storage volume reaches 75% of repository maximum audit record storage capacity.	0	0
APSC-DV-001100 - CAT II Applications categorized as having a moderate or high impact must provide an immediate real-time alert to the SA and ISSO (at a minimum) for all audit failure events.	0	0
APSC-DV-001110 - CAT II The application must alert the ISSO and SA (at a minimum) in the event of an audit processing failure.	0	0
APSC-DV-001120 - CAT II The application must shut down by default upon audit failure (unless availability is an overriding concern).	0	0
APSC-DV-001130 - CAT II The application must provide the capability to centrally review and analyze audit records from multiple components within the system.	0	0
APSC-DV-001140 - CAT II The application must provide the capability to filter audit records for events of interest based upon organization-defined criteria.	0	0
APSC-DV-001150 - CAT II The application must provide an audit reduction capability that supports on-demand reporting requirements.	0	0
APSC-DV-001160 - CAT II The application must provide an audit reduction capability that supports on-demand audit review and analysis.	0	0
APSC-DV-001170 - CAT II The application must provide an audit reduction capability that supports after-the-fact investigations of security incidents.	0	0
APSC-DV-001180 - CAT II The application must provide a report generation capability that supports on-demand audit review and analysis.	0	0

APSC-DV-001190 - CAT II The application must provide a report generation capability that supports on-demand reporting requirements.	0	0
APSC-DV-001200 - CAT II The application must provide a report generation capability that supports after-the-fact investigations of security incidents.	0	0
APSC-DV-001210 - CAT II The application must provide an audit reduction capability that does not alter original content or time ordering of audit records.	0	0
APSC-DV-001220 - CAT II The application must provide a report generation capability that does not alter original content or time ordering of audit records.	0	0
APSC-DV-001250 - CAT II The applications must use internal system clocks to generate time stamps for audit records.	0	0
APSC-DV-001260 - CAT II The application must record time stamps for audit records that can be mapped to Coordinated Universal Time (UTC) or Greenwich Mean Time (GMT).	0	0
APSC-DV-001270 - CAT II The application must record time stamps for audit records that meet a granularity of one second for a minimum degree of precision.	0	0
APSC-DV-001280 - CAT II The application must protect audit information from any type of unauthorized read access.	0	0
APSC-DV-001290 - CAT II The application must protect audit information from unauthorized modification.	0	0
APSC-DV-001300 - CAT II The application must protect audit information from unauthorized deletion.	0	0
APSC-DV-001310 - CAT II The application must protect audit tools from unauthorized access.	0	0
APSC-DV-001320 - CAT II The application must protect audit tools from unauthorized modification.	0	0
APSC-DV-001330 - CAT II The application must protect audit tools from unauthorized deletion.	0	0
APSC-DV-001340 - CAT II The application must back up audit records at least every seven days onto a different system or system component than the system or component being audited.	0	0
APSC-DV-001570 - CAT II The application must electronically verify Personal Identity Verification (PIV) credentials.	0	0
APSC-DV-001350 - CAT II The application must use cryptographic mechanisms to protect the integrity of audit information.	0	0
APSC-DV-001360 - CAT II Application audit tools must be cryptographically hashed.	0	0
APSC-DV-001370 - CAT II The integrity of the audit tools must be validated by checking the files for changes in the cryptographic hash value.	0	0
APSC-DV-001390 - CAT II The application must prohibit user installation of software without explicit privileged status.	0	0
APSC-DV-001410 - CAT II The application must enforce access restrictions associated with changes to application configuration.	0	0
APSC-DV-001420 - CAT II The application must audit who makes configuration changes to the application.	0	0
APSC-DV-001430 - CAT II The application must have the capability to prevent the installation of patches, service packs, or application components without verification the software component has been digitally signed using a certificate that is recognized and approved by the orga	0	0
APSC-DV-001440 - CAT II The applications must limit privileges to change the software resident within software libraries.	0	0
APSC-DV-001460 - CAT II An application vulnerability assessment must be conducted.	0	0
APSC-DV-001480 - CAT II The application must prevent program execution in accordance with organization-defined policies regarding software program usage and restrictions, and/or rules authorizing the terms and conditions of software program usage.	0	0
APSC-DV-001490 - CAT II The application must employ a deny-all, permit-by-exception (whitelist) policy to allow the execution of authorized software programs.	0	0
APSC-DV-001500 - CAT II The application must be configured to disable non-essential capabilities.	0	0
APSC-DV-001510 - CAT II The application must be configured to use only functions, ports, and protocols permitted to it in the PPSM CAL.	0	0

APSC-DV-001520 - CAT II The application must require users to reauthenticate when organization-defined circumstances or situations require reauthentication.	0	0
APSC-DV-001530 - CAT II The application must require devices to reauthenticate when organization-defined circumstances or situations requiring reauthentication.	0	0
APSC-DV-001540 - CAT I The application must uniquely identify and authenticate organizational users (or processes acting on behalf of organizational users).	0	0
APSC-DV-001550 - CAT II The application must use multifactor (Alt. Token) authentication for network access to privileged accounts.	0	0
APSC-DV-001560 - CAT II The application must accept Personal Identity Verification (PIV) credentials.	0	0
APSC-DV-001580 - CAT II The application must use multifactor (e.g., CAC, Alt. Token) authentication for network access to non-privileged accounts.	0	0
APSC-DV-001590 - CAT II The application must use multifactor (Alt. Token) authentication for local access to privileged accounts.	0	0
APSC-DV-001600 - CAT II The application must use multifactor (e.g., CAC, Alt. Token) authentication for local access to non-privileged accounts.	0	0
APSC-DV-001610 - CAT II The application must ensure users are authenticated with an individual authenticator prior to using a group authenticator.	0	0
APSC-DV-001620 - CAT II The application must implement replay-resistant authentication mechanisms for network access to privileged accounts.	0	0
APSC-DV-001630 - CAT II The application must implement replay-resistant authentication mechanisms for network access to non-privileged accounts.	0	0
APSC-DV-001640 - CAT II The application must utilize mutual authentication when endpoint device non-repudiation protections are required by DoD policy or by the data owner.	0	0
APSC-DV-001650 - CAT II The application must authenticate all network connected endpoint devices before establishing any connection.	0	0
APSC-DV-001660 - CAT II Service-Oriented Applications handling non-releasable data must authenticate endpoint devices via mutual SSL/TLS.	0	0
APSC-DV-001670 - CAT II The application must disable device identifiers after 35 days of inactivity unless a cryptographic certificate is used for authentication.	0	0
APSC-DV-001680 - CAT I The application must enforce a minimum 15-character password length.	0	0
APSC-DV-001690 - CAT II The application must enforce password complexity by requiring that at least one upper-case character be used.	0	0
APSC-DV-001700 - CAT II The application must enforce password complexity by requiring that at least one lower-case character be used.	0	0
APSC-DV-001710 - CAT II The application must enforce password complexity by requiring that at least one numeric character be used.	0	0
APSC-DV-001720 - CAT II The application must enforce password complexity by requiring that at least one special character be used.	0	0
APSC-DV-001730 - CAT II The application must require the change of at least 8 of the total number of characters when passwords are changed.	0	0
APSC-DV-001740 - CAT I The application must only store cryptographic representations of passwords.	0	0
APSC-DV-001850 - CAT I The application must not display passwords/PINs as clear text.	0	0
APSC-DV-001750 - CAT I The application must transmit only cryptographically-protected passwords.	0	0
APSC-DV-001760 - CAT II The application must enforce 24 hours/1 day as the minimum password lifetime.	0	0
APSC-DV-001770 - CAT II The application must enforce a 60-day maximum password lifetime restriction.	0	0
APSC-DV-001780 - CAT II The application must prohibit password reuse for a minimum of five generations.	0	0
APSC-DV-001790 - CAT II The application must allow the use of a temporary password for system logons with an immediate change to a permanent password.	0	0

APSC-DV-001795 - CAT II The application password must not be changeable by users other than the administrator or the user with which the password is associated.	0	0
APSC-DV-001800 - CAT II The application must terminate existing user sessions upon account deletion.	0	0
APSC-DV-001820 - CAT I The application, when using PKI-based authentication, must enforce authorized access to the corresponding private key.	0	0
APSC-DV-001830 - CAT II The application must map the authenticated identity to the individual user or group account for PKI-based authentication.	0	0
APSC-DV-001870 - CAT II The application must uniquely identify and authenticate non-organizational users (or processes acting on behalf of non-organizational users).	0	0
APSC-DV-001810 - CAT I The application, when utilizing PKI-based authentication, must validate certificates by constructing a certification path (which includes status information) to an accepted trust anchor.	0	0
APSC-DV-001840 - CAT II The application, for PKI-based authentication, must implement a local cache of revocation data to support path discovery and validation in case of the inability to access revocation information via the network.	0	0
APSC-DV-001860 - CAT II The application must use mechanisms meeting the requirements of applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidance for authentication to a cryptographic module.	0	0
APSC-DV-001880 - CAT II The application must accept Personal Identity Verification (PIV) credentials from other federal agencies.	0	0
APSC-DV-001890 - CAT II The application must electronically verify Personal Identity Verification (PIV) credentials from other federal agencies.	0	0
APSC-DV-002050 - CAT II Applications making SAML assertions must use FIPS-approved random numbers in the generation of SessionIndex in the SAML element AuthnStatement.	0	0
APSC-DV-001900 - CAT II The application must accept FICAM-approved third-party credentials.	0	0
APSC-DV-001910 - CAT II The application must conform to FICAM-issued profiles.	0	0
APSC-DV-001930 - CAT II Applications used for non-local maintenance sessions must audit non-local maintenance and diagnostic sessions for organization-defined auditable events.	0	0
APSC-DV-000310 - CAT III The application must have a process, feature or function that prevents removal or disabling of emergency accounts.	0	0
APSC-DV-001940 - CAT II Applications used for non-local maintenance sessions must implement cryptographic mechanisms to protect the integrity of non-local maintenance and diagnostic communications.	0	0
APSC-DV-001950 - CAT II Applications used for non-local maintenance sessions must implement cryptographic mechanisms to protect the confidentiality of non-local maintenance and diagnostic communications.	0	0
APSC-DV-001960 - CAT II Applications used for non-local maintenance sessions must verify remote disconnection at the termination of non-local maintenance and diagnostic sessions.	0	0
APSC-DV-001970 - CAT II The application must employ strong authenticators in the establishment of non-local maintenance and diagnostic sessions.	0	0
APSC-DV-001980 - CAT II The application must terminate all sessions and network connections when non-local maintenance is completed.	0	0
APSC-DV-001995 - CAT II The application must not be vulnerable to race conditions.	1	1
APSC-DV-002000 - CAT II The application must terminate all network connections associated with a communications session at the end of the session.	0	0
APSC-DV-002010 - CAT II The application must implement NSA-approved cryptography to protect classified information in accordance with applicable federal laws, Executive Orders, directives, policies, regulations, and standards.	0	0
APSC-DV-002020 - CAT II The application must utilize FIPS-validated cryptographic modules when signing application components.	0	0
APSC-DV-002030 - CAT II The application must utilize FIPS-validated cryptographic modules when generating cryptographic hashes.	0	0

APSC-DV-002040 - CAT II The application must utilize FIPS-validated cryptographic modules when protecting unclassified information that requires cryptographic protection.	0	0
APSC-DV-002150 - CAT II The application user interface must be either physically or logically separated from data storage and management interfaces.	0	0
APSC-DV-002210 - CAT II The application must set the HTTPOnly flag on session cookies.*	0	0
APSC-DV-002220 - CAT II The application must set the secure flag on session cookies.	0	0
APSC-DV-002230 - CAT I The application must not expose session IDs.*	0	0
APSC-DV-002240 - CAT I The application must destroy the session ID value and/or cookie on logoff or browser close.	0	0
APSC-DV-002250 - CAT II Applications must use system-generated session identifiers that protect against session fixation.	0	0
APSC-DV-002260 - CAT II Applications must validate session identifiers.	0	0
APSC-DV-002270 - CAT II Applications must not use URL embedded session IDs.	0	0
APSC-DV-002280 - CAT II The application must not re-use or recycle session IDs.	0	0
APSC-DV-002290 - CAT II The application must use the Federal Information Processing Standard (FIPS) 140-2-validated cryptographic modules and random number generator if the application implements encryption, key exchange, digital signature, and hash functionality.	0	0
APSC-DV-002300 - CAT II The application must only allow the use of DoD-approved certificate authorities for verification of the establishment of protected sessions.	0	0
APSC-DV-002310 - CAT I The application must fail to a secure state if system initialization fails, shutdown fails, or aborts fail.	0	0
APSC-DV-002320 - CAT II In the event of a system failure, applications must preserve any information necessary to determine cause of failure and any information necessary to return to operations with least disruption to mission processes.	0	0
APSC-DV-002330 - CAT II The application must protect the confidentiality and integrity of stored information when required by DoD policy or the information owner.	0	0
APSC-DV-002340 - CAT II The application must implement approved cryptographic mechanisms to prevent unauthorized modification of organization-defined information at rest on organization-defined information system components.	0	0
APSC-DV-002350 - CAT II The application must use appropriate cryptography in order to protect stored DoD information when required by the information owner or DoD policy.	0	0
APSC-DV-002360 - CAT II The application must isolate security functions from non-security functions.	0	0
APSC-DV-002370 - CAT II The application must maintain a separate execution domain for each executing process.	0	0
APSC-DV-002380 - CAT II Applications must prevent unauthorized and unintended information transfer via shared system resources.	0	0
APSC-DV-002390 - CAT II XML-based applications must mitigate DoS attacks by using XML filters, parser options, or gateways.	0	0
APSC-DV-002400 - CAT II The application must restrict the ability to launch Denial of Service (DoS) attacks against itself or other information systems.*	1	1
APSC-DV-002410 - CAT II The web service design must include redundancy mechanisms when used with high-availability systems.	0	0
APSC-DV-002420 - CAT II An XML firewall function must be deployed to protect web services when exposed to untrusted networks.	0	0
APSC-DV-002610 - CAT II The application must remove organization-defined software components after updated versions have been installed.	0	0
APSC-DV-002440 - CAT I The application must protect the confidentiality and integrity of transmitted information.	0	0
APSC-DV-002450 - CAT II The application must implement cryptographic mechanisms to prevent unauthorized disclosure of information and/or detect changes to information during transmission unless otherwise protected by alternative physical safeguards, such as, at a minimum, a Prot	0	0

APSC-DV-002460 - CAT II The application must maintain the confidentiality and integrity of information during preparation for transmission.	2	1
APSC-DV-002470 - CAT II The application must maintain the confidentiality and integrity of information during reception.	0	0
APSC-DV-002480 - CAT II The application must not disclose unnecessary information to users.	0	0
APSC-DV-002485 - CAT I The application must not store sensitive information in hidden fields.	0	0
APSC-DV-002490 - CAT I The application must protect from Cross-Site Scripting (XSS) vulnerabilities.	1	1
APSC-DV-002500 - CAT II The application must protect from Cross-Site Request Forgery (CSRF) vulnerabilities.	0	0
APSC-DV-002510 - CAT I The application must protect from command injection.	1	1
APSC-DV-002520 - CAT II The application must protect from canonical representation vulnerabilities.	0	0
APSC-DV-002530 - CAT II The application must validate all input.	0	0
APSC-DV-002540 - CAT I The application must not be vulnerable to SQL Injection.*	0	0
APSC-DV-002550 - CAT I The application must not be vulnerable to XML-oriented attacks.	0	0
APSC-DV-002560 - CAT I The application must not be subject to input handling vulnerabilities.*	12	5
APSC-DV-002570 - CAT II The application must generate error messages that provide information necessary for corrective actions without revealing information that could be exploited by adversaries.*	27	27
APSC-DV-002580 - CAT II The application must reveal error messages only to the ISSO, ISSM, or SA.*	0	0
APSC-DV-002590 - CAT I The application must not be vulnerable to overflow attacks.	0	0
APSC-DV-002630 - CAT II Security-relevant software updates and patches must be kept up to date.	0	0
APSC-DV-002760 - CAT II The application performing organization-defined security functions must verify correct operation of security functions.	0	0
APSC-DV-002900 - CAT II The ISSO must ensure application audit trails are retained for at least 1 year for applications without SAMI data, and 5 years for applications including SAMI data.	0	0
APSC-DV-002770 - CAT II The application must perform verification of the correct operation of security functions: upon system startup and/or restart; upon command by a user with privileged access; and/or every 30 days.	0	0
APSC-DV-002780 - CAT III The application must notify the ISSO and ISSM of failed security verification tests.	0	0
APSC-DV-002870 - CAT II Unsigned Category 1A mobile code must not be used in the application in accordance with DoD policy.	0	0
APSC-DV-002880 - CAT II The ISSO must ensure an account management process is implemented, verifying only authorized users can gain access to the application, and individual accounts designated as inactive, suspended, or terminated are promptly removed.	0	0
APSC-DV-002890 - CAT I Application web servers must be on a separate network segment from the application and database servers if it is a tiered application operating in the DoD DMZ.	0	0
APSC-DV-002910 - CAT II The ISSO must review audit trails periodically based on system documentation recommendations or immediately upon system security events.	0	0
APSC-DV-002920 - CAT II The ISSO must report all suspected violations of IA policies in accordance with DoD information system IA procedures.	0	0
APSC-DV-002930 - CAT II The ISSO must ensure active vulnerability testing is performed.	0	0
APSC-DV-002980 - CAT II New IP addresses, data services, and associated ports used by the application must be submitted to the appropriate approving authority for the organization, which in turn will be submitted through the DoD Ports, Protocols, and Services Management (DoD PPS	0	0
APSC-DV-002950 - CAT II Execution flow diagrams and design documents must be created to show how deadlock and recursion issues in web services are being mitigated.	0	0
APSC-DV-002960 - CAT II The designer must ensure the application does not store configuration and control files in the same directory as user data.	0	0
APSC-DV-002970 - CAT II The ISSO must ensure if a DoD STIG or NSA guide is not available, a third-party	0	0

product will be configured by following available guidance.		
APSC-DV-002990 - CAT II The application must be registered with the DoD Ports and Protocols Database.	0	0
APSC-DV-002995 - CAT II The Configuration Management (CM) repository must be properly patched and STIG compliant.	0	0
APSC-DV-003000 - CAT II Access privileges to the Configuration Management (CM) repository must be reviewed every three months.	0	0
APSC-DV-003010 - CAT II A Software Configuration Management (SCM) plan describing the configuration control and change management process of application objects developed by the organization and the roles and responsibilities of the organization must be created and maintained.	0	0
APSC-DV-003020 - CAT II A Configuration Control Board (CCB) that meets at least every release cycle, for managing the Configuration Management (CM) process must be established.	0	0
APSC-DV-003030 - CAT II The application services and interfaces must be compatible with and ready for IPv6 networks.	0	0
APSC-DV-003040 - CAT II The application must not be hosted on a general purpose machine if the application is designated as critical or high availability by the ISSO.	0	0
APSC-DV-003050 - CAT II A disaster recovery/continuity plan must exist in accordance with DoD policy based on the applications availability requirements.	0	0
APSC-DV-003060 - CAT II Recovery procedures and technical system features must exist so recovery is performed in a secure and verifiable manner. The ISSO will document circumstances inhibiting a trusted recovery.	0	0
APSC-DV-003070 - CAT II Data backup must be performed at required intervals in accordance with DoD policy.	0	0
APSC-DV-003080 - CAT II Back-up copies of the application software or source code must be stored in a fire-rated container or stored separately (offsite).	0	0
APSC-DV-003090 - CAT II Procedures must be in place to assure the appropriate physical and technical protection of the backup and restoration of the application.	0	0
APSC-DV-003100 - CAT II The application must use encryption to implement key exchange and authenticate endpoints prior to establishing a communication channel for key exchange.	0	0
APSC-DV-003110 - CAT I The application must not contain embedded authentication data.	0	0
APSC-DV-003120 - CAT I The application must have the capability to mark sensitive/classified output when required.	0	0
APSC-DV-003130 - CAT III Prior to each release of the application, updates to system, or applying patches; tests plans and procedures must be created and executed.	0	0
APSC-DV-003150 - CAT II At least one tester must be designated to test for security flaws in addition to functional testing.	0	0
APSC-DV-003140 - CAT II Application files must be cryptographically hashed prior to deploying to DoD operational networks.	0	0
APSC-DV-003160 - CAT III Test procedures must be created and at least annually executed to ensure system initialization, shutdown, and aborts are configured to verify the system remains in a secure state.	0	0
APSC-DV-003170 - CAT II An application code review must be performed on the application.	0	0
APSC-DV-003180 - CAT III Code coverage statistics must be maintained for each release of the application.	0	0
APSC-DV-003190 - CAT II Flaws found during a code review must be tracked in a defect tracking system.	0	0
APSC-DV-003200 - CAT II The changes to the application must be assessed for IA and accreditation impact prior to implementation.	0	0
APSC-DV-003210 - CAT II Security flaws must be fixed or addressed in the project plan.	0	0
APSC-DV-003215 - CAT III The application development team must follow a set of coding standards.	0	0
APSC-DV-003220 - CAT III The designer must create and update the Design Document for each release of the application.	0	0
APSC-DV-003230 - CAT II Threat models must be documented and reviewed for each application release	0	0

and updated as required by design and functionality changes or when new threats are discovered.		
APSC-DV-003235 - CAT II The application must not be subject to error handling vulnerabilities.*	0	0
APSC-DV-003250 - CAT I The application must be decommissioned when maintenance or support is no longer available.	0	0
APSC-DV-003236 - CAT II The application development team must provide an application incident response plan.	0	0
APSC-DV-003240 - CAT I All products must be supported by the vendor or the development team.	0	0
APSC-DV-003260 - CAT III Procedures must be in place to notify users when an application is decommissioned.	0	0
APSC-DV-003270 - CAT II Unnecessary built-in application accounts must be disabled.	0	0
APSC-DV-003280 - CAT I Default passwords must be changed.	0	0
APSC-DV-003330 - CAT II The system must alert an administrator when low resource conditions are encountered.	0	0
APSC-DV-003285 - CAT II An Application Configuration Guide must be created and included with the application.	0	0
APSC-DV-003290 - CAT II If the application contains classified data, a Security Classification Guide must exist containing data elements and their classification.	0	0
APSC-DV-003300 - CAT II The designer must ensure uncategorized or emerging mobile code is not used in applications.	0	0
APSC-DV-003310 - CAT II Production database exports must have database administration credentials and sensitive data removed before releasing the export.	0	0
APSC-DV-003320 - CAT II Protections against DoS attacks must be implemented.	0	0
APSC-DV-003340 - CAT III At least one application administrator must be registered to receive update notifications, or security alerts, when automated alerts are available.	0	0
APSC-DV-003360 - CAT III The application must generate audit records when concurrent logons from different workstations occur.	0	0
APSC-DV-003345 - CAT III The application must provide notifications or alerts when product update and security related patches are available.	0	0
APSC-DV-003350 - CAT II Connections between the DoD enclave and the Internet or other public or commercial wide area networks must require a DMZ.	0	0
APSC-DV-003400 - CAT II The Program Manager must verify all levels of program management, designers, developers, and testers receive annual security training pertaining to their job function.	0	0
APSC-DV-000010 - CAT II The application must provide a capability to limit the number of logon sessions per user.	0	0
APSC-DV-000060 - CAT II The application must clear temporary storage and cookies when the session is terminated.	0	0
APSC-DV-000070 - CAT II The application must automatically terminate the non-privileged user session and log off non-privileged users after a 15 minute idle time period has elapsed.	0	0
APSC-DV-000080 - CAT II The application must automatically terminate the admin user session and log off admin users after a 10 minute idle time period is exceeded.	0	0
APSC-DV-000090 - CAT II Applications requiring user access authentication must provide a logoff capability for user initiated communication session.	0	0
APSC-DV-000100 - CAT III The application must display an explicit logoff message to users indicating the reliable termination of authenticated communications sessions.	0	0
APSC-DV-000110 - CAT II The application must associate organization-defined types of security attributes having organization-defined security attribute values with information in storage.	0	0
APSC-DV-000120 - CAT II The application must associate organization-defined types of security attributes having organization-defined security attribute values with information in process.	0	0
APSC-DV-000130 - CAT II The application must associate organization-defined types of security attributes having organization-defined security attribute values with information in transmission.	0	0

APSC-DV-000160 - CAT II The application must implement DoD-approved encryption to protect the confidentiality of remote access sessions.	0	0
APSC-DV-000170 - CAT II The application must implement cryptographic mechanisms to protect the integrity of remote access sessions.	0	0
APSC-DV-000190 - CAT I Messages protected with WS_Security must use time stamps with creation and expiration times.	0	0
APSC-DV-000180 - CAT II Applications with SOAP messages requiring integrity must include the following message elements:-Message ID-Service Request-Timestamp-SAML Assertion (optionally included in messages) and all elements of the message must be digitally signed.	0	0
APSC-DV-000200 - CAT I Validity periods must be verified on all application messages using WS-Security or SAML assertions.	0	0
APSC-DV-000210 - CAT II The application must ensure each unique asserting party provides unique assertion ID references for each SAML assertion.	0	0
APSC-DV-000220 - CAT II The application must ensure encrypted assertions, or equivalent confidentiality protections are used when assertion data is passed through an intermediary, and confidentiality of the assertion data is required when passing through the intermediary.	0	0
APSC-DV-000230 - CAT I The application must use the NotOnOrAfter condition when using the SubjectConfirmation element in a SAML assertion.	0	0
APSC-DV-000240 - CAT I The application must use both the NotBefore and NotOnOrAfter elements or OneTimeUse element when using the Conditions element in a SAML assertion.	0	0
APSC-DV-000250 - CAT II The application must ensure if a OneTimeUse element is used in an assertion, there is only one of the same used in the Conditions element portion of an assertion.	0	0
APSC-DV-000260 - CAT II The application must ensure messages are encrypted when the SessionIndex is tied to privacy data.	0	0
APSC-DV-000290 - CAT II Shared/group account credentials must be terminated when members leave the group.	0	0
APSC-DV-000280 - CAT II The application must provide automated mechanisms for supporting account management functions.	0	0
APSC-DV-000300 - CAT II The application must automatically remove or disable temporary user accounts 72 hours after account creation.	0	0
APSC-DV-000320 - CAT III The application must automatically disable accounts after a 35 day period of account inactivity.	0	0
APSC-DV-000330 - CAT II Unnecessary application accounts must be disabled, or deleted.	0	0
APSC-DV-000420 - CAT II The application must automatically audit account enabling actions.	0	0
APSC-DV-000340 - CAT II The application must automatically audit account creation.	0	0
APSC-DV-000350 - CAT II The application must automatically audit account modification.	0	0
APSC-DV-000360 - CAT II The application must automatically audit account disabling actions.	0	0
APSC-DV-000370 - CAT II The application must automatically audit account removal actions.	0	0
APSC-DV-000380 - CAT III The application must notify System Administrators and Information System Security Officers when accounts are created.	0	0
APSC-DV-000390 - CAT III The application must notify System Administrators and Information System Security Officers when accounts are modified.	0	0
APSC-DV-000400 - CAT III The application must notify System Administrators and Information System Security Officers of account disabling actions.	0	0
APSC-DV-000410 - CAT III The application must notify System Administrators and Information System Security Officers of account removal actions.	0	0
APSC-DV-000430 - CAT III The application must notify System Administrators and Information System Security Officers of account enabling actions.	0	0
APSC-DV-000440 - CAT II Application data protection requirements must be identified and documented.	0	0
APSC-DV-000520 - CAT II The application must audit the execution of privileged functions.	0	0

APSC-DV-000450 - CAT II The application must utilize organization-defined data mining detection techniques for organization-defined data storage objects to adequately detect data mining attempts.	0	0
APSC-DV-000460 - CAT I The application must enforce approved authorizations for logical access to information and system resources in accordance with applicable access control policies.*	0	0
APSC-DV-000470 - CAT II The application must enforce organization-defined discretionary access control policies over defined subjects and objects.*	0	0
APSC-DV-000480 - CAT II The application must enforce approved authorizations for controlling the flow of information within the system based on organization-defined information flow control policies.	0	0
APSC-DV-000490 - CAT II The application must enforce approved authorizations for controlling the flow of information between interconnected systems based on organization-defined information flow control policies.	0	0
APSC-DV-000500 - CAT II The application must prevent non-privileged users from executing privileged functions to include disabling, circumventing, or altering implemented security safeguards/countermeasures.	0	0
APSC-DV-000510 - CAT I The application must execute without excessive account permissions.	0	0
APSC-DV-000530 - CAT I The application must enforce the limit of three consecutive invalid logon attempts by a user during a 15 minute time period.	0	0
APSC-DV-000560 - CAT III The application must retain the Standard Mandatory DoD Notice and Consent Banner on the screen until users acknowledge the usage conditions and take explicit actions to log on for further access.	0	0
APSC-DV-000540 - CAT II The application administrator must follow an approved process to unlock locked user accounts.	0	0
APSC-DV-000550 - CAT III The application must display the Standard Mandatory DoD Notice and Consent Banner before granting access to the application.	0	0
APSC-DV-000570 - CAT III The publicly accessible application must display the Standard Mandatory DoD Notice and Consent Banner before granting access to the application.	0	0
APSC-DV-000580 - CAT III The application must display the time and date of the users last successful logon.	0	0
APSC-DV-000630 - CAT II The application must provide audit record generation capability for the destruction of session IDs.	0	0
APSC-DV-000590 - CAT II The application must protect against an individual (or process acting on behalf of an individual) falsely denying having performed organization-defined actions to be covered by non-repudiation.	0	0
APSC-DV-000600 - CAT II For applications providing audit record aggregation, the application must compile audit records from organization-defined information system components into a system-wide audit trail that is time-correlated with an organization-defined level of tolerance	0	0
APSC-DV-000610 - CAT II The application must provide the capability for organization-identified individuals or roles to change the auditing to be performed on all application components, based on all selectable event criteria within organization-defined time thresholds.	0	0
APSC-DV-000620 - CAT II The application must provide audit record generation capability for the creation of session IDs.	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 2010

Category	Issues Found	Best Fix Locations
A1-Injection	0	0
A2-Cross-Site Scripting (XSS)	0	0
A3-Broken Authentication and Session Management*	0	0
A4-Insecure Direct Object References	1	1
A5-Cross-Site Request Forgery (CSRF)	0	0
A6-Security Misconfiguration	0	0
A7-Insecure Cryptographic Storage	0	0
A8-Failure to Restrict URL Access	0	0
A9-Insufficient Transport Layer Protection	0	0
A10-Unvalidated Redirects and Forwards	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 - MOIS(KISA) Secure Coding 2021

Category	Issues Found	Best Fix Locations
MOIS(KISA) API misuse*	0	0
MOIS(KISA) Code error	1	1
MOIS(KISA) Encapsulation*	0	0
MOIS(KISA) Error processing*	27	27
MOIS(KISA) Security Functions*	23	22
MOIS(KISA) Time and status*	1	1
MOIS(KISA) Verification and representation of input data*	9	8

***** 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 - SANS top 25

Category	Issues Found	Best Fix Locations
SANS top 25*	32	30

* 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 - CWE top 25

Category	Issues Found	Best Fix Locations
CWE top 25*	32	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 - Top Tier

Category	Issues Found	Best Fix Locations
Top Tier	1	1

掃描總結 - OWASP ASVS

Category	Issues Found	Best Fix Locations
V01 Architecture, Design and Threat Modeling*	2	2
V02 Authentication	0	0
V03 Session Management*	0	0
V04 Access Control	21	21
V05 Validation, Sanitization and Encoding*	41	5
V06 Stored Cryptography*	0	0
V07 Error Handling and Logging*	9	3
V08 Data Protection	0	0
V09 Communication*	2	1
V10 Malicious Code	0	0
V11 Business Logic*	0	0
V12 Files and Resources	6	5
V13 API and Web Service	0	0
V14 Configuration*	86	86

* 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 - ASA Mobile Premium

Category	Issues Found	Best Fix Locations
ASA Mobile Premium*	8	2

* 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。

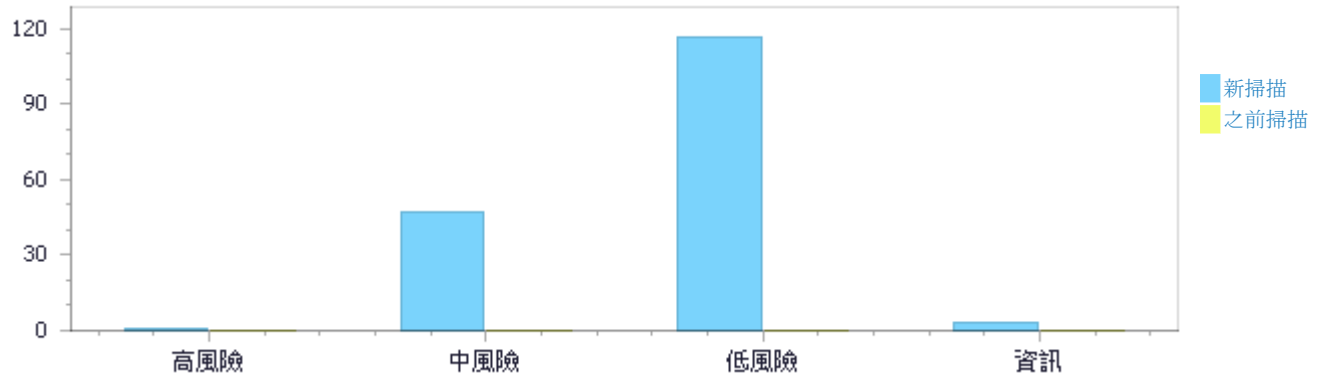
掃描總結 - ASA Premium

Category	Issues Found	Best Fix Locations
ASA Premium*	54	10

***** 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。

掃描結果分佈 專案的首次掃描

	高風險	中風險	低風險	資訊	總共
新問題	1	47	117	3	168
反覆出現的問題	0	0	0	0	0
總共	1	47	117	3	168
已修復的問題	0	0	0	0	0



掃描結果分佈

	高風險	中風險	低風險	資訊	總共
校驗	1	47	117	3	168
不可利用	0	0	0	0	0
確認	0	0	0	0	0
緊急	0	0	0	0	0
推薦不可用	0	0	0	0	0
總共	1	47	117	3	168

掃描結果摘要

漏洞類別	事件	嚴重程度：
Reflected XSS All Clients	1	高風險
Unsafe Object Binding	38	中風險
Absolute Path Traversal	3	中風險
Cleartext Submission of Sensitive Information	2	中風險
Input Path Not Canonicalized	1	中風險

Missing HSTS Header	1	中風險
SSRF	1	中風險
Stored Absolute Path Traversal	1	中風險
Spring Overly Permissive Cross Origin Resource Sharing Policy	57	低風險
Improper Exception Handling	26	低風險
Improper Resource Access Authorization	18	低風險
Log Forging	8	低風險
Incorrect Permission Assignment For Critical Resources	3	低風險
Improper Resource Shutdown or Release	1	低風險
Information Exposure Through an Error Message	1	低風險
Race Condition Format Flaw	1	低風險
Spring Missing Content Security Policy	1	低風險
Spring Missing Expect CT Header	1	低風險
Insufficient Logging of Exceptions	1	資訊
Potential Usage of Vulnerable Log4J	1	資訊
Undocumented API	1	資訊

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
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/bean/dto/RequestDto.java	40
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/bean/ActionInfo.java	30
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/bean/BaseBean.java	29
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	28
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	28
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	7
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java	6
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	5
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java	4
cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/bean/dto/FormDataDto.java	4

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:9

類別

PCI DSS v3.2.1: PCI DSS (3.2.1) - 6.5.7 - Cross-site scripting (XSS)
OWASP Top 10 2013: A3-Cross-Site Scripting (XSS)
FISMA 2014: System And Information Integrity
NIST SP 800-53: SI-15 Information Output Filtering (P0)
OWASP Top 10 2017: A7-Cross-Site Scripting (XSS)
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Verification and representation of input data
OWASP ASVS: V05 Validation, Sanitization and Encoding
OWASP Top 10 2021: A3-Injection
SANS top 25: SANS top 25
ASA Premium: ASA Premium
ASD STIG 5.2: APSC-DV-002490 - CAT I The application must protect from Cross-Site Scripting (XSS) vulnerabilities.
Top Tier: Top Tier

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=1
狀態	新的
Detection Date	6/26/2024 2:23:45 PM

方法uploadFile在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java的第45行使用uploadFile將不受信任的資料嵌入生成的輸出。這些不受信任的資料被嵌入輸出而沒有進行適當的消毒或編碼，使攻擊者能夠將惡意程式碼注入生成的網頁。
攻擊者可以通過簡單地在使用者輸入file中提供修改過的資料，該資料由cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java中的uploadFile方法在第45行讀取。然後該輸入資料直接通過程式碼流到輸出網頁，而不經過任何消毒處理。
這可能會導致反射型跨站腳本（XSS）攻擊。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
行	45	49
物件	file	uploadFile

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
方法	public String uploadFile(@RequestParam("file") MultipartFile file,

```
....
45.  public String uploadFile(@RequestParam("file") MultipartFile file,
....
49.  return attachFileService.uploadFile(actionInfo, file);
```

Unsafe Object Binding

查詢路徑:

Java\公司\Java Medium Threat\Unsafe Object Binding 版本:3

類別

OWASP ASVS: V05 Validation, Sanitization and Encoding
OWASP Top 10 2021: A8-Software and Data Integrity Failures
ASA Premium: ASA Premium

描述

Unsafe Object Binding\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=56
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 64 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java 中 76 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
行	64	78
物件	requestString	save

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public void authorise(@RequestBody String requestString) throws Exception {

```
....
64.  public void authorise(@RequestBody String requestString) throws
Exception {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java

方法	public void save(SignatureDataPo signatureData) {

```
....
78.  signatureDataDao.save(signatureData);
```

Unsafe Object Binding\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=57
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 173 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java 中 76 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
行	173	78
物件	req	save

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
173.  public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody
RequestDto<SubCaseDto> req) {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
方法	public void save(SignatureDataPo signatureData) {

```
....
78.  signatureDataDao.save(signatureData);
```

Unsafe Object Binding\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=58
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 114 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java 中 76 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
行	114	78
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
114.  public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java

方法 public void save(SignatureDataPo signatureData) {

```
....
78.  signatureDataDao.save(signatureData);
```

Unsafe Object Binding\路徑 4:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=59
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 100 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java 中 76 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
行	100	78
物件	requestString	save

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody String requestString) throws Exception {

```
....
100.  public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody
String requestString) throws Exception {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
方法	public void save(SignatureDataPo signatureData) {

```
....
78.  signatureDataDao.save(signatureData);
```

Unsafe Object Binding\路徑 5:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=60
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 79 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java 中 76 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
行	79	78
物件	req	save

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public void authorise(@RequestBody RequestDto<BatchAuthDto> req) throws Exception {

```
....
79.  public void authorise(@RequestBody RequestDto<BatchAuthDto> req)
throws Exception {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
方法	public void save(SignatureDataPo signatureData) {

```
....
78.  signatureDataDao.save(signatureData);
```

Unsafe Object Binding\路徑 6:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=61
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 49 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java 中 76 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java
行	49	78
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ActionInfo submit(@RequestBody String requestString) throws Exception {

```
....
49.  public ActionInfo submit(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/SignatureHelper.java

方法 public void save(SignatureDataPo signatureData) {

```
....
78.  signatureDataDao.save(signatureData);
```

Unsafe Object Binding\路徑 7:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=62
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 中 66 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java 中 112 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java
行	66	118
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public void updateCompanyForDelete(@RequestBody RequestDto<String> req) { companyService.updateCompanyForDelete(req.getActionInfo(), req.getBody()); }

```
....
66.  public void updateCompanyForDelete(@RequestBody RequestDto<String>
req) { companyService.updateCompanyForDelete(req.getActionInfo(),
req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

方法 public void updateCompanyForDelete(ActionInfo actionInfo, String companyId) {

```
....
118.  eFormCompanyDao.save(oldCompany);
```

Unsafe Object Binding\路徑 8:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=63
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 79 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 636 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	79	646

物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void authorise(@RequestBody RequestDto<BatchAuthDto> req) throws Exception {

```
....
79.  public void authorise(@RequestBody RequestDto<BatchAuthDto> req)
throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private void saveFileMetadata(ActionInfo actionInfo, List<FileMetadataPo> files) {

```
....
646.  fileMetadataDao.save(po);
```

Unsafe Object Binding\路徑 9:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=64
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 49 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 636 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	49	646
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ActionInfo submit(@RequestBody String requestString) throws Exception {

```
....
49.  public ActionInfo submit(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private void saveFileMetadata(ActionInfo actionInfo, List<FileMetadataPo> files) {

```
....
646.  fileMetadataDao.save(po);
```

Unsafe Object Binding\路徑 10:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=65
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 64 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 636 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	64	646
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void authorise(@RequestBody String requestString) throws Exception {

```
....
64.  public void authorise(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private void saveFileMetadata(ActionInfo actionInfo, List<FileMetadataPo> files) {

```
....
646.  fileMetadataDao.save(po);
```

Unsafe Object Binding\路徑 11:

嚴重程度： 中風險

結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=66
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 100 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 636 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	100	646
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody String requestString) throws Exception {

```
....
100.  public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody
String requestString) throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private void saveFileMetadata(ActionInfo actionInfo, List<FileMetadataPo> files) {

```
....
646.  fileMetadataDao.save(po);
```

Unsafe Object Binding\路徑 12:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=67
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 173 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 522 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	173	586
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
173.  public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody
RequestDto<SubCaseDto> req) {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private String createCase(String formId, String creatUserid, String targetUserid, String targetName, List<FormDataPo> formDataList) {

```
....
586.  processControllerDao.save(pc);
```

Unsafe Object Binding\路徑 13:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=68
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 114 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 522 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	114	586
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
114.  public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req)  {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private String createCase(String formId, String creatUserid, String targetUserid, String targetName, List<FormDataPo> formDataList) {

```
....
586.  processControllerDao.save(pc);
```

Unsafe Object Binding\路徑 14:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=69
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 79 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java 中 146 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java
行	79	165
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void authorise(@RequestBody RequestDto<BatchAuthDto> req) throws Exception {

```
....
79.  public void authorise(@RequestBody RequestDto<BatchAuthDto> req)
throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java

方法 public void comment(ActionInfo actionInfo, String actionType) {

```
....
165.  caseCommentDao.save(cp);
```

Unsafe Object Binding\路徑 15:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=70
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 49 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java 中 146 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java
行	49	165
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ActionInfo submit(@RequestBody String requestString) throws Exception {

```
....
49.  public ActionInfo submit(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java

方法 public void comment(ActionInfo actionInfo, String actionType) {

```
....
165.  caseCommentDao.save(cp);
```

Unsafe Object Binding\路徑 16:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=71
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 64 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java 中 146 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java
行	64	165
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void authorise(@RequestBody String requestString) throws Exception {

```
....
64.  public void authorise(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java

方法 public void comment(ActionInfo actionInfo, String actionType) {

```
....
165.  caseCommentDao.save(cp);
```

Unsafe Object Binding\路徑 17:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=72
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 89 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java 中 146 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java
行	89	165
物件	req	save

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public void reject(@RequestBody RequestDto<Object> req) {

```
....
89.  public void reject(@RequestBody RequestDto<Object> req) {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/CamundaHelper.java
方法	public void comment(ActionInfo actionInfo, String actionType) {

```
....
165.  caseCommentDao.save(cp);
```

Unsafe Object Binding\路徑 18:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=73
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 173 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 476 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	173	504
物件	req	save

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
173.  public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody
RequestDto<SubCaseDto> req) {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法	public ResponseDto<SubCaseDto> autoCreateCase(ActionInfo actionInfo, SubCaseDto subCaseDto) {

```
....
504.  subCaseDao.save(subCase);
```

Unsafe Object Binding\路徑 19:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=74
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 195 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 608 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	195	612
物件	req	save

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public ProcessControllerPo addUrgentNote(@RequestBody RequestDto<ProcessControllerPo> req) {

```
....
195.  public ProcessControllerPo addUrgentNote(@RequestBody
RequestDto<ProcessControllerPo> req) {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
方法	public ProcessControllerPo addUrgentNote(ActionInfo actionInfo, ProcessControllerPo po) {

```
....
612.  processControllerDao.save(po);
```

Unsafe Object Binding\路徑 20:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathi

	d=75
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 79 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 216 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	79	223
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void authorise(@RequestBody RequestDto<BatchAuthDto> req) throws Exception {

```
....
79.  public void authorise(@RequestBody RequestDto<BatchAuthDto> req)
throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private void saveCaseData(String instanceId, List<FormDataPo> formDatas, ActionInfo actionInfo) {

```
....
223.  formDataDao.save(colData);
```

Unsafe Object Binding\路徑 21:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=76
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 49 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 216 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-	cicd-form-backend-

	checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	49	223
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ActionInfo submit(@RequestBody String requestString) throws Exception {

```
....
49.  public ActionInfo submit(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private void saveCaseData(String instanceId, List<FormDataPo> formDatas, ActionInfo actionInfo) {

```
....
223.  formDataDao.save(colData);
```

Unsafe Object Binding\路徑 22:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=77
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 64 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 216 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	64	223
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void authorise(@RequestBody String requestString) throws Exception {

```
....
64.  public void authorise(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private void saveCaseData(String instanceId, List<FormDataPo> formDatas, ActionInfo actionInfo) {

```
....
223.  formDataDao.save(colData);
```

Unsafe Object Binding\路徑 23:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=78
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 100 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 216 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	100	223
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody String requestString) throws Exception {

```
....
100.  public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody
String requestString) throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法	private void saveCaseData(String instanceId, List<FormDataPo> formDatas, ActionInfo actionInfo) {

```
....
223.  formDataDao.save(colData);
```

Unsafe Object Binding\路徑 24:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=79
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java 中 71 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 中 142 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	71	150
物件	req	save

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
方法	public void deleteFile(@RequestBody RequestDto<String> req) throws Exception {

```
....
71.  public void deleteFile(@RequestBody RequestDto<String> req) throws
Exception {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
方法	public void deleteFile(ActionInfo actionInfo, String uuid) throws IOException {

```
....
150.  fileMetadataDao.save(po);
```

Unsafe Object Binding\路徑 25:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=80
狀態	新的

Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 173 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 522 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	173	558
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
173.  public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody
RequestDto<SubCaseDto> req) {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private String createCase(String formId, String creatUserid, String targetUserid, String targetName, List<FormDataPo> formDataList) {

```
....
558.  formDataDao.save(data);
```

Unsafe Object Binding\路徑 26:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=81
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 114 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 522 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/c	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/s

	ontroller/CaseActionController.java	ervice/impl/CaseActionServiceImpl.java
行	114	558
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
114.  public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req)  {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private String createCase(String formId, String creatUserid, String targetUserid, String targetName, List<FormDataPo> formDataList) {

```
....
558.  formDataDao.save(data);
```

Unsafe Object Binding\路徑 27:

嚴重程度： 中風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=82
狀態 新的
Detection Date 6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 中 63 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java 中 87 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
行	63	98
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java

方法 public void updateCompanyUser(@RequestBody RequestDto<EFormCompanyUserPo> req) { companyUserService.updateCompanyUser(req.getActionInfo(), req.getBody()); }

```
....
63.  public void updateCompanyUser(@RequestBody
RequestDto<EFormCompanyUserPo> req) {
companyUserService.updateCompanyUser(req.getActionInfo(),
req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java

方法 public void updateCompanyUser(ActionInfo actionInfo, EFormCompanyUserPo companyUser) {

```
....
98.  eFormCompanyUserDao.save(oldCompanyUser);
```

Unsafe Object Binding\路徑 28:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=83
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java 中 54 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java 中 75 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java
行	54	87
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java

方法 public void updateEmployee(@RequestBody RequestDto<EFormUserPo> req) { employeeService.updateEmployee(req.getActionInfo(), req.getBody()); }

```
....
54.  public void updateEmployee(@RequestBody RequestDto<EFormUserPo>
req) { employeeService.updateEmployee(req.getActionInfo(),
req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java

方法	public void updateEmployee(ActionInfo actionInfo, EFormUserPo employee) {

```
....
87.  eFormUserDao.save(oldEmployee);
```

Unsafe Object Binding\路徑 29:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=84
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java 中 51 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java 中 54 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java
行	51	64
物件	req	save

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
方法	public void addEmployee(@RequestBody RequestDto<EFormUserPo> req) { employeeService.addEmployee(req.getActionInfo(), req.getBody()); }

```
....
51.  public void addEmployee(@RequestBody RequestDto<EFormUserPo> req) {
employeeService.addEmployee(req.getActionInfo(), req.getBody()); }
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java
方法	public void addEmployee(ActionInfo actionInfo, EFormUserPo employee) {

```
....
64.  eFormUserDao.save(employee);
```

Unsafe Object Binding\路徑 30:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=85
狀態	新的

Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 中 60 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java 中 64 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
行	60	71
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java

方法 public void addCompanyUser(@RequestBody RequestDto<EFormCompanyUserPo> req) { companyUserService.addCompanyUser(req.getActionInfo(), req.getBody()); }

```
....
60.  public void addCompanyUser(@RequestBody
RequestDto<EFormCompanyUserPo> req) {
companyUserService.addCompanyUser(req.getActionInfo(), req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java

方法 public void addCompanyUser(ActionInfo actionInfo, EFormCompanyUserPo companyUser) {

```
....
71.  eFormCompanyUserDao.save(companyUser);
```

Unsafe Object Binding\路徑 31:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=86
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 中 60 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java 中 64 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

行	60	71
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public void addCompany(@RequestBody RequestDto<EFormCompanyPo> req) { companyService.addCompany(req.getActionInfo(), req.getBody()); }

```
....
60.  public void addCompany(@RequestBody RequestDto<EFormCompanyPo> req)
{ companyService.addCompany(req.getActionInfo(), req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

方法 public void addCompany(ActionInfo actionInfo, EFormCompanyPo company) {

```
....
71.  eFormCompanyDao.save(company);
```

Unsafe Object Binding\路徑 32:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=87
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 49 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 100 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	49	128
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ActionInfo submit(@RequestBody String requestString) throws Exception {

```
....
49.  public ActionInfo submit(@RequestBody String requestString) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public ActionInfo submit(ActionInfo actionInfo, FormDataDto<FormDataPo> formData) {

```
....
128.  processControllerDao.save(pc);
```

Unsafe Object Binding\路徑 33:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=88
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java 中 47 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FlowGroupRelationServiceImpl.java 中 50 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FlowGroupRelationServiceImpl.java
行	47	53
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java

方法 public void updateFlowRelation(@RequestBody RequestDto<List<FlowGroupRelationPo>> req) throws Exception {

```
....
47.  public void updateFlowRelation(@RequestBody
RequestDto<List<FlowGroupRelationPo>> req) throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FlowGroupRelationServiceImpl.java

方法 public void updateFlowRelation(ActionInfo actionInfo, List<FlowGroupRelationPo> flowGroupRelationList) {

```
....
53.  flowGroupRelationDao.save(po);
```

Unsafe Object Binding\路徑 34:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=89
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 114 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 356 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	114	378
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
114.  public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public ResponseDto<SubCaseDto> saveSubCase(ActionInfo actionInfo, SubCaseDto subCaseDto) {

```
....
378.  subCaseDao.save(subCase);
```

Unsafe Object Binding\路徑 35:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=90
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 中 63 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java 中 87 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java
行	63	100
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public void updateCompany(@RequestBody RequestDto<EFormCompanyPo> req) { companyService.updateCompany(req.getActionInfo(), req.getBody()); }

```
....
63.  public void updateCompany(@RequestBody RequestDto<EFormCompanyPo>
req) { companyService.updateCompany(req.getActionInfo(), req.getBody());
}
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

方法 public void updateCompany(ActionInfo actionInfo, EFormCompanyPo company) {

```
....
100.  eFormCompanyDao.save(oldCompany);
```

Unsafe Object Binding\路徑 36:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=91
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 100 行的 requestString 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 330 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

行	100	348
物件	requestString	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody String requestString) throws Exception {

```
....
100.  public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody
String requestString) throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public ResponseDto<FormDataDto<FormDataPo>> save(ActionInfo actionInfo, FormDataDto<FormDataPo> formData) {

```
....
348.  processControllerDao.save(pc);
```

Unsafe Object Binding\路徑 37:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=92
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 中 66 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java 中 110 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
行	66	116
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java

方法 public void updateCompanyUserForDelete(@RequestBody RequestDto<String> req) { companyUserService.updateCompanyUserForDelete(req.getActionInfo(), req.getBody()); }

```
....
66.  public void updateCompanyUserForDelete(@RequestBody
RequestDto<String> req) {
companyUserService.updateCompanyUserForDelete(req.getActionInfo(),
req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java

方法 public void updateCompanyUserForDelete(ActionInfo actionInfo, String companyUserId) {

```
....
116.  eFormCompanyUserDao.save(oldCompanyUser);
```

Unsafe Object Binding\路徑 38:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=93
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 中 89 行的 req 可能意外地允許設置 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 中 316 行的 save 的值。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	89	326
物件	req	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void reject(@RequestBody RequestDto<Object> req) {

```
....
89.  public void reject(@RequestBody RequestDto<Object> req) {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void reject(ActionInfo actionInfo) {

```
....
326.  processControllerDao.save(pc);
```

Absolute Path Traversal

查詢路徑:

Java\Cx\Java Medium Threat\Absolute Path Traversal 版本:2

類別

PCI DSS v3.2.1: PCI DSS (3.2.1) - 6.5.8 - Improper access control
OWASP Top 10 2013: A4-Insecure Direct Object References
OWASP Top 10 2017: A5-Broken Access Control
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Verification and representation of input data
OWASP ASVS: V12 Files and Resources
OWASP Top 10 2021: A3-Injection
SANS top 25: SANS top 25
ASA Premium: ASA Premium
ASD STIG 5.2: APSC-DV-002560 - CAT I The application must not be subject to input handling vulnerabilities.

描述

Absolute Path Traversal\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=2
狀態	新的
Detection Date	6/26/2024 2:23:45 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java 第 45 行的 uploadFile 方法從 file 元素取得動態資料。該元素的值隨後流經程式碼，最終在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 57 行的 uploadFile 方法中用於本地硬碟訪問的檔案路徑。這可能導致路徑遍歷漏洞。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	45	73
物件	file	copy

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java

方法 public String uploadFile(@RequestParam("file") MultipartFile file,

```
....
45.  public String uploadFile(@RequestParam("file") MultipartFile file,
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java

方法 public String uploadFile(ActionInfo actionInfo, MultipartFile file) throws IOException {

```
....
73.  Files.copy(file.getInputStream(), Paths.get(directory.toString(),
uuid), StandardCopyOption.REPLACE_EXISTING);
```

Absolute Path Traversal\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=3
狀態	新的
Detection Date	6/26/2024 2:23:45 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java 第 60 行的 downloadFile 方法從 req 元素取得動態資料。該元素的值隨後流經程式碼，最終在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 99 行的 downloadFile 方法中用於本地硬碟訪問的檔案路徑。這可能導致路徑遍歷漏洞。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	60	102
物件	req	readAllBytes

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java

方法 public ResponseEntity<byte[]> downloadFile(@RequestBody RequestDto<String> req) throws Exception {

```
....
60.  public ResponseEntity<byte[]> downloadFile(@RequestBody
RequestDto<String> req) throws Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java

方法 public ResponseEntity<byte[]> downloadFile(ActionInfo actionInfo, String uuid) throws IOException {

```
....
102.  byte[] fileContent = Files.readAllBytes(filePath);
```

Absolute Path Traversal\路徑 3:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=4
狀態	新的
Detection Date	6/26/2024 2:23:45 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java 第 71 行的 deleteFile 方法從 req 元素取得動態資料。該元素的值隨後流經程式碼，最終在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 161 行的 deleteFileByName 方法中用於本地硬碟訪問的檔案路徑。這可能導致路徑遍歷漏洞。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	71	165
物件	req	delete

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java

方法 public void deleteFile(@RequestBody RequestDto<String> req) throws Exception {

```
....
71.  public void deleteFile(@RequestBody RequestDto<String> req) throws
Exception {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java

方法 private void deleteFileByName(String uuid) throws IOException {

```
....
165.  Files.delete(filePath);
```

Cleartext Submission of Sensitive Information

查詢路徑:

Java\Cx\Java Medium Threat\Cleartext Submission of Sensitive Information 版本:6

類別

PCI DSS v3.2.1: PCI DSS (3.2.1) - 6.5.4 - Insecure communications
OWASP Top 10 2013: A6-Sensitive Data Exposure
FISMA 2014: Configuration Management
NIST SP 800-53: SC-8 Transmission Confidentiality and Integrity (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure
OWASP Top 10 API: API7-Security Misconfiguration

MOIS(KISA) Secure Coding 2021: MOIS(KISA) Security Functions
OWASP ASVS: V09 Communication
OWASP Top 10 2021: A2-Cryptographic Failures
SANS top 25: SANS top 25
ASA Premium: ASA Premium
ASD STIG 5.2: APSC-DV-002460 - CAT II The application must maintain the confidentiality and integrity of information during preparation for transmission.

描述

Cleartext Submission of Sensitive Information\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=7
狀態	新的
Detection Date	6/26/2024 2:23:45 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java 第 237 行的潛在敏感個人資料 cipher， 透過非安全通道被傳送到 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java 第 183 行的 httpConnect 中。這可能暴露這些個人資料並允許其被竊取。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java
行	255	208
物件	cipher	getBytes

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

方法 private JsonObject createBodyData(String type, String cipher) {

```
....
255.  returnBody.addProperty("cipher", cipher);
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

方法 public StringBuilder httpConnect(String type, String cipher) {

```
....
208.  os.write(bodyData.toString().getBytes(StandardCharsets.UTF_8));
```

Cleartext Submission of Sensitive Information\路徑 2:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=8
狀態	新的

Detection Date	6/26/2024 2:23:45 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java 第 237 行的潛在敏感個人資料 cipher， 透過非安全通道被傳送到 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java 第 183 行的 httpConnect 中。這可能暴露這些個人資料並允許其被竊取。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java
行	255	208
物件	cipher	write

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

方法 private JsonObject createBodyData(String type, String cipher) {

```
....
255.  returnBody.addProperty("cipher", cipher);
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

方法 public StringBuilder httpConnect(String type, String cipher) {

```
....
208.  os.write(bodyData.toString().getBytes(StandardCharsets.UTF_8));
```

Input Path Not Canonicalized

查詢路徑:

Java\Cx\Java Medium Threat\Input Path Not Canonicalized 版本:7

類別

FISMA 2014: System And Information Integrity
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Mobile Top 10 2016: M7-Client Code Quality
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Verification and representation of input data
OWASP ASVS: V12 Files and Resources
OWASP Top 10 2021: A4-Insecure Design
SANS top 25: SANS top 25
ASD STIG 5.2: APSC-DV-002560 - CAT I The application must not be subject to input handling vulnerabilities.

描述

Input Path Not Canonicalized\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=5

狀態	新的
Detection Date	6/26/2024 2:23:45 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 99 行的 downloadFile 方法從 filePath 元素取得動態資料。該元素的值隨後流經程式碼，最終在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 99 行的 downloadFile 方法中用於本地硬碟訪問的檔案路徑。這可能導致路徑遍歷漏洞。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	102	102
物件	filePath	readAllBytes

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
方法	public ResponseEntity<byte[]> downloadFile(ActionInfo actionInfo, String uuid) throws IOException {

```
....
102.  byte[] fileContent = Files.readAllBytes(filePath);
```

Missing HSTS Header

查詢路徑:

Java\公司\Java Medium Threat\Missing HSTS Header 版本:2

類別

OWASP ASVS: V14 Configuration
OWASP Top 10 2021: A7-Identification and Authentication Failures
ASA Premium: ASA Premium

描述

Missing HSTS Header\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=6
狀態	新的
Detection Date	6/26/2024 2:23:45 PM

網頁應用程式沒有設定HTTP強制安全傳輸技術(HTTP Strict Transport Security，簡稱HSTS) header 導致容易被受到攻擊。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

行	46	46
物件	HsmHelper	HsmHelper

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java
方法 @Component

```
....
46.  @Component
```

SSRF

查詢路徑:
Java\Cx\Java Medium Threat\SSRF 版本:3

類別

FISMA 2014: System And Information Integrity
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A5-Broken Access Control
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Verification and representation of input data
OWASP ASVS: V05 Validation, Sanitization and Encoding
OWASP Top 10 2021: A10-Server-Side Request Forgery
ASA Premium: ASA Premium

描述

SSRF\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=9
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

應用程式使用cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java文件中的第183行的write傳送請求至遠端伺服器，攻擊者能夠透過cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java文件中的第273行的readLine來控制該請求傳送URL或其他資料。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java
行	280	208
物件	readLine	write

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java
方法 private StringBuilder readResponseContent(InputStream inputStream) throws IOException {

```
....
280.  while ((line = in.readLine()) != null) {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java
方法	public StringBuilder httpConnect(String type, String cipher) {

```
....
208.  os.write(bodyData.toString().getBytes(StandardCharsets.UTF_8));
```

Stored Absolute Path Traversal

查詢路徑:

Java\Cx\Java Medium Threat\Stored Absolute Path Traversal 版本:5

類別

PCI DSS v3.2.1: PCI DSS (3.2.1) - 6.5.8 - Improper access control
OWASP Top 10 2013: A4-Insecure Direct Object References
OWASP Top 10 2017: A5-Broken Access Control
OWASP Top 10 2010: A4-Insecure Direct Object References
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Verification and representation of input data
OWASP ASVS: V12 Files and Resources
OWASP Top 10 2021: A3-Injection
SANS top 25: SANS top 25

描述

Stored Absolute Path Traversal\路徑 1:

嚴重程度：	中風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=10
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 99 行的 downloadFile 方法從 filePath 元素取得動態資料。該元素的值隨後流經程式碼，最終在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 99 行的 downloadFile 方法中用於本地硬碟訪問的檔案路徑。這可能導致路徑遍歷漏洞。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	102	102
物件	filePath	readAllBytes

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
方法	public ResponseEntity<byte[]> downloadFile(ActionInfo actionInfo, String uuid) throws IOException {

```
....
102.  byte[] fileContent = Files.readAllBytes(filePath);
```

Spring Overly Permissive Cross Origin Resource Sharing Policy

查詢路徑:

Java\Cx\Java Spring\Spring Overly Permissive Cross Origin Resource Sharing Policy 版本:2

類別

OWASP ASVS: V14 Configuration
OWASP Top 10 2021: A7-Identification and Authentication Failures

描述

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=109
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java 文件第 45 行上的 uploadFile 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
行	45	45
物件	uploadFile	uploadFile

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
方法	public String uploadFile(@RequestParam("file") MultipartFile file,

```
....
45.  public String uploadFile(@RequestParam("file") MultipartFile file,
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=110
狀態	新的

Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java 文件第 60 行上的 downloadFile 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
行	60	60
物件	downloadFile	downloadFile

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java

方法 public ResponseEntity<byte[]> downloadFile(@RequestBody RequestDto<String> req) throws Exception {

```
....
60.  public ResponseEntity<byte[]> downloadFile(@RequestBody
RequestDto<String> req) throws Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=111
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java 文件第 71 行上的 deleteFile 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
行	71	71
物件	deleteFile	deleteFile

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java

方法 public void deleteFile(@RequestBody RequestDto<String> req) throws Exception {

```
....
71.  public void deleteFile(@RequestBody RequestDto<String> req) throws
Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=112
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 49 行上的 submit 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	49	49
物件	submit	submit

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ActionInfo submit(@RequestBody String requestString) throws Exception {

```
....
49.  public ActionInfo submit(@RequestBody String requestString) throws
Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=113
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 64 行上的 authorise 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	64	64
物件	authorise	authorise

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public void authorise(@RequestBody String requestString) throws Exception {

```
....
64.  public void authorise(@RequestBody String requestString) throws
Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=114
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 79 行上的 authorise 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	79	79
物件	authorise	authorise

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public void authorise(@RequestBody RequestDto<BatchAuthDto> req) throws Exception {

```
....
79.  public void authorise(@RequestBody RequestDto<BatchAuthDto> req)
throws Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=115
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 89 行上的 reject 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/c	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/c

	ontroller/CaseActionController.java	ontroller/CaseActionController.java
行	89	89
物件	reject	reject

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void reject(@RequestBody RequestDto<Object> req) {

```
....
89.  public void reject(@RequestBody RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=116
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 100 行上的 save 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	100	100
物件	save	save

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody String requestString) throws Exception {

```
....
100.  public ResponseDto<FormDataDto<FormDataPo>> save(@RequestBody
String requestString) throws Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 9:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=117
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 114 行上的 saveSubCase 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	114	114
物件	saveSubCase	saveSubCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
114.  public void saveSubCase(@RequestBody RequestDto<SubCaseDto> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 10:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=118
狀態 新的
Detection Date 6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 124 行上的 delete 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	124	124
物件	delete	delete

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public void delete(@RequestBody RequestDto<Object> req) {

```
....
124.  public void delete(@RequestBody RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 11:

嚴重程度： 低風險

結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=119
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 135 行上的 deleteSubCase 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	135	135
物件	deleteSubCase	deleteSubCase

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public void deleteSubCase(@RequestBody RequestDto<SubCasePo> req) {

```
....
135.  public void deleteSubCase(@RequestBody RequestDto<SubCasePo> req)
{
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 12:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=120
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 147 行上的 fetch 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	147	147
物件	fetch	fetch

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
方法	public ResponseDto<FormDataDto<FormDataPo>> fetch(@RequestBody String requestString) throws Exception {

```
....
147.  public ResponseDto<FormDataDto<FormDataPo>> fetch(@RequestBody
String requestString) throws Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 13:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=121
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 162 行上的 view 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	162	162
物件	view	view

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<FormDataDto<FormDataPo>> view(@RequestBody RequestDto<Object> req) {

```
....
162.  public ResponseDto<FormDataDto<FormDataPo>> view(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 14:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=122
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 173 行上的 autoCreateCase 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	173	173

物件	autoCreateCase	autoCreateCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody RequestDto<SubCaseDto> req) {

```
....
173.  public ResponseDto<SubCaseDto> autoCreateCase(@RequestBody
RequestDto<SubCaseDto> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 15:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=123
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 184 行上的 viewSubCaseStatus 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	184	184
物件	viewSubCaseStatus	viewSubCaseStatus

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public List<SubCasePo> viewSubCaseStatus(@RequestBody RequestDto<Object> req) {

```
....
184.  public List<SubCasePo> viewSubCaseStatus(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 16:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=124
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 195 行上的 addUrgentNote 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	195	195
物件	addUrgentNote	addUrgentNote

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public ProcessControllerPo addUrgentNote(@RequestBody RequestDto<ProcessControllerPo> req) {

```
....
195.  public ProcessControllerPo addUrgentNote(@RequestBody
RequestDto<ProcessControllerPo> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 17:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=125
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java 文件第 206 行上的 checkIsNeedDesignatedUser 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java
行	206	206
物件	checkIsNeedDesignatedUser	checkIsNeedDesignatedUser

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseActionController.java

方法 public Boolean checkIsNeedDesignatedUser(@RequestBody RequestDto<Object> req) {

```
....
206.  public Boolean checkIsNeedDesignatedUser(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 18:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=126
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java 文件第 39 行上的 getCaseList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
行	39	39
物件	getCaseList	getCaseList

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
方法	public ResponseDto<List<ProcessControllerPo>> getCaseList(@RequestBody RequestDto<String> req) {

```
....
39.  public ResponseDto<List<ProcessControllerPo>>
getCaseList(@RequestBody RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 19:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=127
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java 文件第 49 行上的 getDashboardCaseCount 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
行	49	49
物件	getDashboardCaseCount	getDashboardCaseCount

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java

方法	public List<Long> getDashboardCaseCount(@RequestBody RequestDto<String> req) {

```
....
49.  public List<Long> getDashboardCaseCount(@RequestBody
RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 20:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=128
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java 文件第 59 行上的 getCommentList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
行	59	59
物件	getCommentList	getCommentList

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
方法	public List<CommentInfo> getCommentList(@RequestBody RequestDto<String> req) {

```
....
59.  public List<CommentInfo> getCommentList(@RequestBody
RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 21:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=129
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java 文件第 70 行上的 findAvailableCase 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java

行	70	70
物件	findAvailableCase	findAvailableCase

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
方法	public List<ProcessControllerPo> findAvailableCase(@RequestBody RequestDto<CaseSelecterQueryDto> req) {

```
....
70.  public List<ProcessControllerPo> findAvailableCase(@RequestBody
RequestDto<CaseSelecterQueryDto> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 22:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=130
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java 文件第 80 行上的 countVisitorCase 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
行	80	80
物件	countVisitorCase	countVisitorCase

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CaseEnqueryController.java
方法	public Long countVisitorCase(@RequestBody RequestDto<String> req) {

```
....
80.  public Long countVisitorCase(@RequestBody RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 23:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=131
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 文件第 37 行上的 getCompanyList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
行	37	37
物件	getCompanyList	getCompanyList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public List<EFormCompanyPo> getCompanyList(@RequestBody RequestDto<Object> req) {

```
....
37.  public List<EFormCompanyPo> getCompanyList(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 24:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=132
狀態 新的
Detection Date 6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 文件第 46 行上的 getMaxCompanyId 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
行	46	46
物件	getMaxCompanyId	getMaxCompanyId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public RequestDto<String> getMaxCompanyId(@RequestBody RequestDto<Object> req) {

```
....
46.  public RequestDto<String> getMaxCompanyId(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 25:

嚴重程度： 低風險

結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=133
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 文件第 55 行上的 selectCompanyName 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
行	55	55
物件	selectCompanyName	selectCompanyName

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
方法	public RequestDto<String> selectCompanyName(@RequestBody RequestDto<String> req) {

```
....
55.  public RequestDto<String> selectCompanyName(@RequestBody
RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 26:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=134
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 文件第 60 行上的 addCompany 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
行	60	60
物件	addCompany	addCompany

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
方法	public void addCompany(@RequestBody RequestDto<EFormCompanyPo> req) { companyService.addCompany(req.getActionInfo(), req.getBody()); }

```
....
60.  public void addCompany(@RequestBody RequestDto<EFormCompanyPo> req)
{ companyService.addCompany(req.getActionInfo(), req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 27:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=135
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 文件第 63 行上的 updateCompany 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
行	63	63
物件	updateCompany	updateCompany

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public void updateCompany(@RequestBody RequestDto<EFormCompanyPo> req) { companyService.updateCompany(req.getActionInfo(), req.getBody()); }

```
....
63.  public void updateCompany(@RequestBody RequestDto<EFormCompanyPo>
req) { companyService.updateCompany(req.getActionInfo(), req.getBody());
}
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 28:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=136
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 文件第 66 行上的 updateCompanyForDelete 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

行	66	66
物件	updateCompanyForDelete	updateCompanyForDelete

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public void updateCompanyForDelete(@RequestBody RequestDto<String> req) { companyService.updateCompanyForDelete(req.getActionInfo(), req.getBody()); }

```
....
66.  public void updateCompanyForDelete(@RequestBody RequestDto<String>
req) { companyService.updateCompanyForDelete(req.getActionInfo(),
req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 29:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=137
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java 文件第 69 行上的 deleteCompany 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java
行	69	69
物件	deleteCompany	deleteCompany

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public void deleteCompany(@RequestBody RequestDto<String> req) { companyService.deleteCompany(req.getBody()); }

```
....
69.  public void deleteCompany(@RequestBody RequestDto<String> req) {
companyService.deleteCompany(req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 30:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=138
狀態	新的

Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 文件第 37 行上的 getCompanyUserList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
行	37	37
物件	getCompanyUserList	getCompanyUserList

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
方法	public List<EFormCompanyUserPo> getCompanyUserList(@RequestBody RequestDto<Object> req) {

```
....
37.  public List<EFormCompanyUserPo> getCompanyUserList(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 31:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=139
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 文件第 46 行上的 getMaxCompanyUserId 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
行	46	46
物件	getMaxCompanyUserId	getMaxCompanyUserId

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
方法	public RequestDto<String> getMaxCompanyUserId(@RequestBody RequestDto<Object> req) {

```
....
46.  public RequestDto<String> getMaxCompanyUserId(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 32:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=140
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 文件第 55 行上的 selectCompanyUserName 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
行	55	55
物件	selectCompanyUserName	selectCompanyUserName

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java

方法 public RequestDto<String> selectCompanyUserName(@RequestBody RequestDto<String> req) {

```
....
55.  public RequestDto<String> selectCompanyUserName(@RequestBody
RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 33:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=141
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 文件第 60 行上的 addCompanyUser 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java

行	60	60
物件	addCompanyUser	addCompanyUser

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
方法	public void addCompanyUser(@RequestBody RequestDto<EFormCompanyUserPo> req) { companyUserService.addCompanyUser(req.getActionInfo(), req.getBody()); }

```
....
60.  public void addCompanyUser(@RequestBody
RequestDto<EFormCompanyUserPo> req) {
companyUserService.addCompanyUser(req.getActionInfo(), req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 34:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=142
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 文件第 63 行上的 updateCompanyUser 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
行	63	63
物件	updateCompanyUser	updateCompanyUser

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
方法	public void updateCompanyUser(@RequestBody RequestDto<EFormCompanyUserPo> req) { companyUserService.updateCompanyUser(req.getActionInfo(), req.getBody()); }

```
....
63.  public void updateCompanyUser(@RequestBody
RequestDto<EFormCompanyUserPo> req) {
companyUserService.updateCompanyUser(req.getActionInfo(),
req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 35:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=143

狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 文件第 66 行上的 updateCompanyUserForDelete 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
行	66	66
物件	updateCompanyUserForDelete	updateCompanyUserForDelete

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
方法	public void updateCompanyUserForDelete(@RequestBody RequestDto<String> req) { companyUserService.updateCompanyUserForDelete(req.getActionInfo(), req.getBody()); }

```
....
66.  public void updateCompanyUserForDelete(@RequestBody
RequestDto<String> req) {
companyUserService.updateCompanyUserForDelete(req.getActionInfo(),
req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 36:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=144
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java 文件第 69 行上的 deleteCompanyUser 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
行	69	69
物件	deleteCompanyUser	deleteCompanyUser

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java
方法	public void deleteCompanyUser(@RequestBody RequestDto<String> req) { companyUserService.deleteCompanyUser(req.getBody()); }

```
....
69.  public void deleteCompanyUser(@RequestBody RequestDto<String> req)
{ companyUserService.deleteCompanyUser(req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 37:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=145
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/DepartmentController.java 文件第 36 行上的 getEmployeeList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/DepartmentController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/DepartmentController.java
行	36	36
物件	getEmployeeList	getEmployeeList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/DepartmentController.java

方法 public List<DepartmentPo> getEmployeeList(@RequestBody RequestDto<Object> req) {

```
....
36.  public List<DepartmentPo> getEmployeeList(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 38:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=146
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java 文件第 36 行上的 getEmployeeList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
行	36	36

物件	getEmployeeList	getEmployeeList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java

方法 public List<EFormUserPo> getEmployeeList(@RequestBody RequestDto<Object> req) {

```
....
36.  public List<EFormUserPo> getEmployeeList(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 39:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=147
狀態 新的
Detection Date 6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java 文件第 46 行上的 selectUserName 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
行	46	46
物件	selectUserName	selectUserName

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java

方法 public RequestDto<String> selectUserName(@RequestBody RequestDto<String> req) {

```
....
46.  public RequestDto<String> selectUserName(@RequestBody
RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 40:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=148
狀態 新的
Detection Date 6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java 文件第 51 行上的 addEmployee 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
行	51	51
物件	addEmployee	addEmployee

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java

方法 public void addEmployee(@RequestBody RequestDto<EFormUserPo> req) { employeeService.addEmployee(req.getActionInfo(), req.getBody()); }

```
....
51.  public void addEmployee(@RequestBody RequestDto<EFormUserPo> req) {
employeeService.addEmployee(req.getActionInfo(), req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 41:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=149
狀態 新的
Detection Date 6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java 文件第 54 行上的 updateEmployee 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
行	54	54
物件	updateEmployee	updateEmployee

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java

方法 public void updateEmployee(@RequestBody RequestDto<EFormUserPo> req) { employeeService.updateEmployee(req.getActionInfo(), req.getBody()); }

```
....
54.  public void updateEmployee(@RequestBody RequestDto<EFormUserPo>
req) { employeeService.updateEmployee(req.getActionInfo(),
req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 42:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=150
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java 文件第 57 行上的 deleteEmployee 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
行	57	57
物件	deleteEmployee	deleteEmployee

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
方法	public void deleteEmployee(@RequestBody RequestDto<String> req) { employeeService.deleteEmployee(req.getBody()); }

```
....
57.  public void deleteEmployee(@RequestBody RequestDto<String> req) {
employeeService.deleteEmployee(req.getBody()); }
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 43:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=151
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java 文件第 37 行上的 searchFlowRelation 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java
行	37	37
物件	searchFlowRelation	searchFlowRelation

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java

方法	public List<FlowGroupRelationPo> searchFlowRelation(@RequestBody RequestDto<String> req) throws Exception {

```
....
37.  public List<FlowGroupRelationPo> searchFlowRelation(@RequestBody
RequestDto<String> req) throws Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 44:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=152
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java 文件第 47 行上的 updateFlowRelation 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java
行	47	47
物件	updateFlowRelation	updateFlowRelation

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FlowRelationController.java
方法	public void updateFlowRelation(@RequestBody RequestDto<List<FlowGroupRelationPo>> req) throws Exception {

```
....
47.  public void updateFlowRelation(@RequestBody
RequestDto<List<FlowGroupRelationPo>> req) throws Exception {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 45:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=153
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FormTemplateController.java 文件第 35 行上的 getFormById 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/c	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/c

	ontroller/FormTemplateController.java	ontroller/FormTemplateController.java
行	35	35
物件	getFormById	getFormById

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FormTemplateController.java

方法 public FormDataDto<FormTemplatePo> getFormById(@RequestBody RequestDto<String> req) {

```
....
35.  public FormDataDto<FormTemplatePo> getFormById(@RequestBody
RequestDto<String> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 46:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=154
狀態 新的
Detection Date 6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java 文件第 38 行上的 getEmployeeFunctionList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java
行	38	38
物件	getEmployeeFunctionList	getEmployeeFunctionList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java

方法 public Map<String, List<FunctionCodePo>> getEmployeeFunctionList(@RequestBody RequestDto<Object> req) {

```
....
38.  public Map<String, List<FunctionCodePo>>
getEmployeeFunctionList(@RequestBody RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 47:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=155

狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java 文件第 48 行上的 getVisitorFunctionList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java
行	48	48
物件	getVisitorFunctionList	getVisitorFunctionList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java

方法 public Map<String, List<FunctionCodePo>> getVisitorFunctionList(@RequestBody RequestDto<Object> req) {

```
....
48.  public Map<String, List<FunctionCodePo>>
getVisitorFunctionList(@RequestBody RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 48:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=156
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java 文件第 60 行上的 getFunctionType 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java
行	60	60
物件	getFunctionType	getFunctionType

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/FunctionCodeController.java

方法 public List<FunctionCodeDto> getFunctionType(@RequestBody RequestDto<Object> req) {

```
....
60.  public List<FunctionCodeDto> getFunctionType(@RequestBody
RequestDto<Object> req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 49:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=157
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java 文件第 36 行上的 getGroupList 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java
行	36	36
物件	getGroupList	getGroupList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java

方法 public List<GroupDto> getGroupList(@RequestBody RequestDto<Object> req) {

```
....
36.  public List<GroupDto> getGroupList(@RequestBody RequestDto<Object>
req) {
```

Spring Overly Permissive Cross Origin Resource Sharing Policy\路徑 50:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=158
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java 文件第 45 行上的 addGroup 方法設置了過度寬鬆的 CORS 訪問控制來源標頭。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java
行	45	45

物件	addGroup	addGroup

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java
方法	public void addGroup(@RequestBody RequestDto<GroupDto> req) {

```
....
45.  public void addGroup(@RequestBody RequestDto<GroupDto> req) {
```

Improper Exception Handling

查詢路徑:

Java\Cx\Java Low Visibility\Improper Exception Handling 版本:1

類別

PCI DSS v3.2.1: PCI DSS (3.2.1) - 6.5.5 - Improper error handling
NIST SP 800-53: SC-5 Denial of Service Protection (P1)
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Error processing
OWASP ASVS: V14 Configuration
OWASP Top 10 2021: A4-Insecure Design
ASD STIG 5.2: APSC-DV-002570 - CAT II The application must generate error messages that provide information necessary for corrective actions without revealing information that could be exploited by adversaries.

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=11
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getReport04001Data在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java第67 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java
行	120	120
物件	query	query

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java
方法	public List<Report04001Vo> getReport04001Data(List<String> instanceIds) {

```
....
120.  return namedParameterJdbcTemplate.query(sql.toString(), paramMap,
BeanPropertyRowMapper.newInstance(Report04001Vo.class));
```

Improper Exception Handling\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=12
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getReportData在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java第123 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java
行	170	170
物件	query	query

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java

方法 private <T> List<T> getReportData(List<String> instanceIds, String[] pivotColumns, String[] beanColumns, String[] joinColumns, Class<T> clazz) {

```
....
170.  return namedParameterJdbcTemplate.query(sql.toString(), paramMap,
BeanPropertyRowMapper.newInstance(clazz));
```

Improper Exception Handling\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=13
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getGroupList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java第39 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/s	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/s

	ervice/impl/GroupServiceImpl.java	ervice/impl/GroupServiceImpl.java
行	41	41
物件	list	list

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java
方法 public List<GroupDto> getGroupList() {

```
....
41.  List<Group> groupList = identityService.createGroupQuery().list();
```

Improper Exception Handling\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=14
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getDepartmentList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/DepartmentServiceImpl.java第35 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/DepartmentServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/DepartmentServiceImpl.java
行	36	36
物件	findAllSortByDepartmentId	findAllSortByDepartmentId

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/DepartmentServiceImpl.java
方法 public List<DepartmentPo> getDepartmentList() {

```
....
36.  return departmentDao.findAllSortByDepartmentId();
```

Improper Exception Handling\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=15
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getCompanyList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java第40 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java
行	41	41
物件	findAllSortByCompanyId	findAllSortByCompanyId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

方法 public List<EFormCompanyPo> getCompanyList() {

```
....
41.  return eFormCompanyDao.findAllSortByCompanyId();
```

Improper Exception Handling\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=16
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getMaxCompanyId在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java第50 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java
行	53	53
物件	findMaxCompanyId	findMaxCompanyId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

方法 public RequestDto<String> getMaxCompanyId(ActionInfo actionInfo) {

```
....
53.  result.setBody(eFormCompanyDao.findMaxCompanyId());
```

Improper Exception Handling\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=17
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法addCompany在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java第64 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java
行	65	65
物件	findMaxCompanyId	findMaxCompanyId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

方法 public void addCompany(ActionInfo actionInfo, EFormCompanyPo company) {

```
....
65.  int maxCompanyId =
Integer.parseInt(eFormCompanyDao.findMaxCompanyId());
```

Improper Exception Handling\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=18
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getCompanyUserList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java第40 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
行	41	41
物件	findAllSortByCompanyUserId	findAllSortByCompanyUserId

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
方法	public List<EFormCompanyUserPo> getCompanyUserList() {

```
....
41.  return eFormCompanyUserDao.findAllSortByCompanyUserId();
```

Improper Exception Handling\路徑 9:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=19
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getMaxCompanyUserId在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java第50 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
行	53	53
物件	findMaxCompanyUserId	findMaxCompanyUserId

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
方法	public RequestDto<String> getMaxCompanyUserId(ActionInfo actionInfo) {

```
....
53.  result.setBody(eFormCompanyUserDao.findMaxCompanyUserId());
```

Improper Exception Handling\路徑 10:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=20
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法addCompanyUser在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java第64 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	cicd-form-backend-	cicd-form-backend-

	checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java	checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
行	65	65
物件	findMaxCompanyUserId	findMaxCompanyUserId

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
方法 public void addCompanyUser(ActionInfo actionInfo, EFormCompanyUserPo companyUser) {

```
....
65.  int maxCompanyUserId =
Integer.parseInt(eFormCompanyUserDao.findMaxCompanyUserId());
```

Improper Exception Handling\路徑 11:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=21
狀態 新的
Detection Date 6/26/2024 2:23:46 PM

方法getEmployeeList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java第44 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java
行	45	45
物件	findAllSortByEmployeeId	findAllSortByEmployeeId

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java
方法 public List<EFormUserPo> getEmployeeList() {

```
....
45.  return eFormUserDao.findAllSortByEmployeeId();
```

Improper Exception Handling\路徑 12:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=22
狀態 新的

Detection Date	6/26/2024 2:23:46 PM

方法getFormDataByinstanceId在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第445 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	454	454
物件	findUploadFile	findUploadFile

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 private FormDataDto<FormDataPo> getFormDataByinstanceId(String instanceId, String formId) {

```
....
454.
formData.setAttachFileList(fileMetadataDao.findUploadFile(instanceId));
```

Improper Exception Handling\路徑 13:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=23
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法delete在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第389 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	402	402
物件	updateFileStatusByInstanceId	updateFileStatusByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void delete(ActionInfo actionInfo) {

```
....
402.
fileMetadataDao.updateFileStatusByInstanceId(actionInfo.getInstanceId(),
DataStatusConstant.DELETED);
```

Improper Exception Handling\路徑 14:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=24
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getFormById在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java第36 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java
行	38	38
物件	findByFormId	findByFormId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java

方法 public FormDataDto<FormTemplatePo> getFormById(String formId) {

```
....
38.  result.setColumnDataList(formTemplateDao.findByFormId(formId));
```

Improper Exception Handling\路徑 15:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=25
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法getEmployeeFunctionList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java第30 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java

行	31	31
物件	findEmployeeFunctionList	findEmployeeFunctionList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java

方法 public Map<String, List<FunctionCodePo>> getEmployeeFunctionList() {

```
....
31.  return
classifyFuncionCode(functionCodeDao.findEmployeeFunctionList());
```

Improper Exception Handling\路徑 16:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=26
狀態 新的
Detection Date 6/26/2024 2:23:46 PM

方法getVisitorFunctionList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java第35 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java
行	36	36
物件	findVisitorFunctionList	findVisitorFunctionList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FunctionCodeServiceImpl.java

方法 public Map<String, List<FunctionCodePo>> getVisitorFunctionList() {

```
....
36.  return
classifyFuncionCode(functionCodeDao.findVisitorFunctionList());
```

Improper Exception Handling\路徑 17:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=27
狀態 新的
Detection Date 6/26/2024 2:23:46 PM

方法authorise在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第250 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	282	282
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void authorise(ActionInfo actionInfo, FormDataDto<FormDataPo> formData) {

```
....
282.
processControllerDao.updateCaseByInstanceId(actionInfo.getInstanceId(),
processUserId,
```

Improper Exception Handling\路徑 18:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=28
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法authorise在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第250 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	287	287
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void authorise(ActionInfo actionInfo, FormDataDto<FormDataPo> formData) {

```
....
287.
processControllerDao.updateCaseByInstanceId(actionInfo.getInstanceId(),
processUserId,
```

Improper Exception Handling\路徑 19:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=29
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法delete在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第389 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	392	392
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void delete(ActionInfo actionInfo) {

```
....
392.
processControllerDao.updateCaseByInstanceId(actionInfo.getInstanceId(),
actionInfo.getUserId(),
```

Improper Exception Handling\路徑 20:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=30
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法deleteSubCase在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第406 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	cicd-form-backend-	cicd-form-backend-

	checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	408	408
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void deleteSubCase(ActionInfo actionInfo, SubCasePo subCase) {

```
....
408.
processControllerDao.updateCaseByInstanceId(subCase.getSubInstanceId(),
actionInfo.getUserId(),
```

Improper Exception Handling\路徑 21:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=31
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法fetch在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第418 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	424	424
物件	countCanFectchCase	countCanFectchCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public ResponseDto<FormDataDto<FormDataPo>> fetch(ActionInfo actionInfo, List<String> instanceIds) {

```
....
424.  if (processControllerDao.countCanFectchCase(instanceIds) > 0) {
```

Improper Exception Handling\路徑 22:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathi

	d=32
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法countProcessList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java第164 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
行	165	165
物件	countProcessingList	countProcessingList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java

方法 public Long countProcessList(String userId) {

```
....
165.  return processControllerDao.countProcessingList(userId);
```

Improper Exception Handling\路徑 23:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=33
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法countVisitorCaseList在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java第156 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
行	157	157
物件	countVisitorCaseList	countVisitorCaseList

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java

方法 private Long countVisitorCaseList(String userName) {

```
....
157.  return processControllerDao.countVisitorCaseList(userName);
```

Improper Exception Handling\路徑 24:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=34
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法findAvailableCase在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java第169 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
行	171	171
物件	findAvailableCase	findAvailableCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java

方法 public List<ProcessControllerPo> findAvailableCase(ActionInfo actionInfo, CaseSelecterQueryDto dto) {

```
....
171.  return processControllerDao.findAvailableCase(instanceIds,
dto.getFormId(), dto.getFlowActions());
```

Improper Exception Handling\路徑 25:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=35
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

方法deleteSubCase在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第406 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

行	413	413
物件	deleteUnSendSubCase	deleteUnSendSubCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void deleteSubCase(ActionInfo actionInfo, SubCasePo subCase) {

```
....
413.  subCaseDao.deleteUnSendSubCase(subCase.getMainInstanceId(),
subCase.getEmployeeId());
```

Improper Exception Handling\路徑 26:

嚴重程度： 低風險
結果狀態： 校驗
線上結果 http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=36
狀態 新的
Detection Date 6/26/2024 2:23:46 PM

方法deleteSubCase在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java第406 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	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	411	411
物件	deleteSendedSubCase	deleteSendedSubCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void deleteSubCase(ActionInfo actionInfo, SubCasePo subCase) {

```
....
411.  subCaseDao.deleteSendedSubCase(subCase.getSubInstanceId());
```

Improper Resource Access Authorization

查詢路徑:

Java\Cx\Java Low Visibility\Improper Resource Access Authorization 版本:11

類別

FISMA 2014: Identification And Authentication
NIST SP 800-53: AC-3 Access Enforcement (P1)
CWE top 25: CWE top 25

MOIS(KISA) Secure Coding 2021: MOIS(KISA) Security Functions
OWASP ASVS: V04 Access Control
OWASP Top 10 2021: A1-Broken Access Control
SANS top 25: SANS top 25

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=37
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 的第 99 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	102	102
物件	readAllBytes	readAllBytes

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java

方法 public ResponseEntity<byte[]> downloadFile(ActionInfo actionInfo, String uuid) throws IOException {

```
....
102.  byte[] fileContent = Files.readAllBytes(filePath);
```

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=38
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java 的第 131 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

行	142	142
物件	readLine	readLine

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

方法 private PrivateKey createPrivateKey() throws NoSuchAlgorithmException, InvalidKeySpecException {

```
....
142.  while ((strTemp = bf.readLine()) != null) {
```

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=39
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 406 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	408	408
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void deleteSubCase(ActionInfo actionInfo, SubCasePo subCase) {

```
....
408.
processControllerDao.updateCaseByInstanceId(subCase.getSubInstanceId(),
actionInfo.getUserId(),
```

Improper Resource Access Authorization\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=40
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 406 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	411	411
物件	deleteSendedSubCase	deleteSendedSubCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void deleteSubCase(ActionInfo actionInfo, SubCasePo subCase) {

```
....
411.  subCaseDao.deleteSendedSubCase(subCase.getSubInstanceId());
```

Improper Resource Access Authorization\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=41
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 406 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	413	413
物件	deleteUnSendSubCase	deleteUnSendSubCase

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void deleteSubCase(ActionInfo actionInfo, SubCasePo subCase) {

```
....
413.  subCaseDao.deleteUnSendSubCase(subCase.getMainInstanceId(),
subCase.getEmployeeId());
```

Improper Resource Access Authorization\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=42
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java 的第 67 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java
行	120	120
物件	query	query

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java
方法	public List<Report04001Vo> getReport04001Data(List<String> instanceIds) {

```
....
120.  return namedParameterJdbcTemplate.query(sql.toString(), paramMap,
BeanPropertyRowMapper.newInstance(Report04001Vo.class));
```

Improper Resource Access Authorization\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=43
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java 的第 123 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java
行	170	170
物件	query	query

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/dao/impl/ReportDaoImpl.java

方法	private <T> List<T> getReportData(List<String> instanceIds, String[] pivotColumns, String[] beanColumns, String[] joinColumns, Class<T> clazz) {

```
....
170.  return namedParameterJdbcTemplate.query(sql.toString(), paramMap,
BeanPropertyRowMapper.newInstance(clazz));
```

Improper Resource Access Authorization\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=44
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java 的第 156 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
行	157	157
物件	countVisitorCaseList	countVisitorCaseList

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
方法	private Long countVisitorCaseList(String userName) {

```
....
157.  return processControllerDao.countVisitorCaseList(userName);
```

Improper Resource Access Authorization\路徑 9:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=45
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java 的第 164 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/s	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/s

	ervice/impl/CaseEnqueryServiceImpl.java	ervice/impl/CaseEnqueryServiceImpl.java
行	165	165
物件	countProcessingList	countProcessingList

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
方法 public Long countProcessList(String userId) {

```
....
165.  return processControllerDao.countProcessingList(userId);
```

Improper Resource Access Authorization\路徑 10:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=46
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 250 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	282	282
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
方法 public void authorise(ActionInfo actionInfo, FormDataDto<FormDataPo> formData) {

```
....
282.
processControllerDao.updateCaseByInstanceId(actionInfo.getInstanceId(),
processUserId,
```

Improper Resource Access Authorization\路徑 11:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=47
狀態	新的

Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 250 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	287	287
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void authorise(ActionInfo actionInfo, FormDataDto<FormDataPo> formData) {

```
....
287.
processControllerDao.updateCaseByInstanceId(actionInfo.getInstanceId(),
processUserId,
```

Improper Resource Access Authorization\路徑 12:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=48
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 389 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	392	392
物件	updateCaseByInstanceId	updateCaseByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void delete(ActionInfo actionInfo) {

```
....
392.
processControllerDao.updateCaseByInstanceId(actionInfo.getInstanceId(),
actionInfo.getUserId(),
```

Improper Resource Access Authorization\路徑 13:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=49
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 389 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	402	402
物件	updateFileStatusByInstanceId	updateFileStatusByInstanceId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java

方法 public void delete(ActionInfo actionInfo) {

```
....
402.
fileMetadataDao.updateFileStatusByInstanceId(actionInfo.getInstanceId(),
DataStatusConstant.DELETED);
```

Improper Resource Access Authorization\路徑 14:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=50
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 418 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-	cicd-form-backend-

	checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	424	424
物件	countCanFectchCase	countCanFectchCase

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
方法	public ResponseDto<FormDataDto<FormDataPo>> fetch(ActionInfo actionInfo, List<String> instanceIds) {

```
....
424.  if (processControllerDao.countCanFectchCase(instanceIds) > 0) {
```

Improper Resource Access Authorization\路徑 15:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=51
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java 的第 445 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
行	454	454
物件	findUploadFile	findUploadFile

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseActionServiceImpl.java
方法	private FormDataDto<FormDataPo> getFormDataByinstanceId(String instanceId, String formId) {

```
....
454.
formData.setAttachFileList(fileMetadataDao.findUploadFile(instanceId));
```

Improper Resource Access Authorization\路徑 16:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathi

	d=52
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 的第 57 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	73	73
物件	copy	copy

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
方法	public String uploadFile(ActionInfo actionInfo, MultipartFile file) throws IOException {

```
....
73.  Files.copy(file.getInputStream(), Paths.get(directory.toString(),
uuid), StandardCopyOption.REPLACE_EXISTING);
```

Improper Resource Access Authorization\路徑 17:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=53
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java 的第 169 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
行	171	171
物件	findAvailableCase	findAvailableCase

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CaseEnqueryServiceImpl.java
方法	public List<ProcessControllerPo> findAvailableCase(ActionInfo actionInfo, CaseSelecterQueryDto dto) {

```
....
171.  return processControllerDao.findAvailableCase(instanceIds,
dto.getFormId(), dto.getFlowActions());
```

Improper Resource Access Authorization\路徑 18:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=54
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在檔案 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java 的第 36 行執行了 I/O 操作，卻沒有進行授權檢查。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java
行	38	38
物件	findByFormId	findByFormId

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/FormTemplateServiceImpl.java

方法 public FormDataDto<FormTemplatePo> getFormById(String formId) {

```
....
38.  result.setColumnDataList(formTemplateDao.findByFormId(formId));
```

Log Forging

查詢路徑:

Java\Cx\Java Low Visibility\Log Forging 版本:4

類別

PCI DSS v3.2.1: PCI DSS (3.2.1) - 6.5.1 - Injection flaws - particularly SQL injection
FISMA 2014: System And Information Integrity
NIST SP 800-53: AU-9 Protection of Audit Information (P1)
OWASP Top 10 2017: A1-Injection
OWASP Mobile Top 10 2016: M7-Client Code Quality
OWASP ASVS: V07 Error Handling and Logging
OWASP Top 10 2021: A9-Security Logging and Monitoring Failures
ASA Mobile Premium: ASA Mobile Premium
ASA Premium: ASA Premium
ASD STIG 5.2: APSC-DV-002560 - CAT I The application must not be subject to input handling vulnerabilities.

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=98
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java的45行的uploadFile方法，從元素file中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java中57行的uploadFile方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	45	76
物件	file	info

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
方法	public String uploadFile(@RequestParam("file") MultipartFile file,

```
....
45.  public String uploadFile(@RequestParam("file") MultipartFile file,
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
方法	public String uploadFile(ActionInfo actionInfo, MultipartFile file) throws IOException {

```
....
76.  logger.info("檔案上傳成功 uuid: " + uuid);
```

Log Forging\路徑 2:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=99
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java的69行的

deleteCompanyUser方法，從元素req中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java中127行的deleteCompanyUser方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java
行	69	135
物件	req	info

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyUserController.java

方法 public void deleteCompanyUser(@RequestBody RequestDto<String> req) { companyUserService.deleteCompanyUser(req.getBody()); }

```
....
69.  public void deleteCompanyUser(@RequestBody RequestDto<String> req)
{ companyUserService.deleteCompanyUser(req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyUserServiceImpl.java

方法 public void deleteCompanyUser(String companyUserId) {

```
....
135.  logger.info("Company User deleted with ID: " + companyUserId);
```

Log Forging\路徑 3:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=100
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java的69行的deleteCompany方法，從元素req中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java中129行的deleteCompany方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-	cicd-form-backend-

	checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java	checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java
行	69	137
物件	req	info

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/CompanyController.java

方法 public void deleteCompany(@RequestBody RequestDto<String> req) { companyService.deleteCompany(req.getBody()); }

```
....
69.  public void deleteCompany(@RequestBody RequestDto<String> req) {
companyService.deleteCompany(req.getBody()); }
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/CompanyServiceImpl.java

方法 public void deleteCompany(String companyId) {

```
....
137.  logger.info("Company deleted with ID: " + companyId);
```

Log Forging\路徑 4:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=101
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java的64行的updateGroup方法，從元素req中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java中76行的updateGroup方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java
行	64	84
物件	req	info

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java
方法	public void updateGroup(@RequestBody RequestDto<GroupDto> req) {

```
....
64.  public void updateGroup(@RequestBody RequestDto<GroupDto> req) {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java
方法	public void updateGroup(GroupDto groupDto) {

```
....
84.  logger.info("Group updated with ID: " + groupDto.getId());
```

Log Forging\路徑 5:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=102
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java的71行的deleteFile方法，從元素req中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java中161行的deleteFileByName方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	71	166
物件	req	info

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
方法	public void deleteFile(@RequestBody RequestDto<String> req) throws Exception {

```
....
71.  public void deleteFile(@RequestBody RequestDto<String> req) throws
Exception {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
方法	private void deleteFileByName(String uuid) throws IOException {

```
....
166.  logger.info("檔案已成功刪除 uuid: " + uuid);
```

Log Forging\路徑 6:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=103
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java的57行的deleteEmployee方法，從元素req中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java中97行的deleteEmployee方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java
行	57	107
物件	req	info

代碼片斷

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/EmployeeController.java
方法	public void deleteEmployee(@RequestBody RequestDto<String> req) { employeeService.deleteEmployee(req.getBody()); }

```
....
57.  public void deleteEmployee(@RequestBody RequestDto<String> req) {
employeeService.deleteEmployee(req.getBody()); }
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/EmployeeServiceImpl.java
方法	public void deleteEmployee(String employeeId) {

```
....
107.  logger.info("Employee deleted with ID: " + employeeId);
```

Log Forging\路徑 7:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=104
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java的71行的deleteFile方法，從元素req中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java中161行的deleteFileByName方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	71	164
物件	req	info

代碼片斷	
檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
方法	public void deleteFile(@RequestBody RequestDto<String> req) throws Exception {

```
....
71.  public void deleteFile(@RequestBody RequestDto<String> req) throws
Exception {
```

檔案名稱	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
方法	private void deleteFileByName(String uuid) throws IOException {

```
....
164.  logger.info("無法刪除檔案 uuid: " + uuid);
```

Log Forging\路徑 8:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=105
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

位於cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java的73行的deleteGroup方法，從元素req中獲取了使用者輸入。該元素的值在程式碼中未經適當的消毒或驗證即被傳遞

，最終在cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java中106行的deleteGroup方法中用於寫入稽核日誌。
這可能會導致稽核日誌偽造攻擊(Log Forging)。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java
行	73	111
物件	req	info

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/GroupController.java

方法 public void deleteGroup(@RequestBody RequestDto<String> req) {

```
....
73.  public void deleteGroup(@RequestBody RequestDto<String> req) {
```

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/GroupServiceImpl.java

方法 public void deleteGroup(String groupId) {

```
....
111.  logger.info("Group deleted with ID: " + groupId);
```

Incorrect Permission Assignment For Critical Resources

查詢路徑:

Java\Cx\Java Low Visibility\Incorrect Permission Assignment For Critical Resources 版本:6

類別

FISMA 2014: Access Control
NIST SP 800-53: AC-3 Access Enforcement (P1)
OWASP Top 10 2017: A6-Security Misconfiguration
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Security Functions
OWASP ASVS: V04 Access Control
OWASP Top 10 2021: A4-Insecure Design
SANS top 25: SANS top 25

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=95
狀態 新的

Detection Date	6/26/2024 2:23:46 PM

檔案系統中由 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 57 行的 directory 建立一個具有潛在危險權限的檔案。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	61	61
物件	directory	directory

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java

方法 public String uploadFile(ActionInfo actionInfo, MultipartFile file) throws IOException {

```
....
61.  Path directory = Paths.get(path, dateString);
```

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=96
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

檔案系統中由 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 99 行的 filePath 建立一個具有潛在危險權限的檔案。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	101	101
物件	filePath	filePath

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java

方法 public ResponseEntity<byte[]> downloadFile(ActionInfo actionInfo, String uuid) throws IOException {

```
....
101.  Path filePath = Paths.get(this.getFilePath(uuid));
```

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=97
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

檔案系統中由 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java 第 161 行的 filePath 建立一個具有潛在危險權限的檔案。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java
行	163	163
物件	filePath	filePath

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/service/impl/AttachFileServiceImpl.java

方法 private void deleteFileByName(String uuid) throws IOException {

```
....
163.  Path filePath = Paths.get(this.getFilePath(uuid));
```

Improper Resource Shutdown or Release

查詢路徑:

Java\Cx\Java Low Visibility\Improper Resource Shutdown or Release 版本:9

類別

NIST SP 800-53: SC-5 Denial of Service Protection (P1)
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Code error
ASD STIG 5.2: APSC-DV-002400 - CAT II The application must restrict the ability to launch Denial of Service (DoS) attacks against itself or other information systems.

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=55
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

應用程式在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java 中的 httpConnect method 定義和初始化 183 中的 getOutputStream 物件。該物件封裝(Encapsulation)了有限的資源，例如：開啟的檔案串流、資料庫連線、網路串流等，但這些資源並非在所有情況下都會被正確的關閉和釋放

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java
行	207	208
物件	getOutputStream	write

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/HsmHelper.java

方法 public StringBuilder httpConnect(String type, String cipher) {

```
....
207.  OutputStream os = con.getOutputStream();
208.  os.write(bodyData.toString().getBytes(StandardCharsets.UTF_8));
```

Information Exposure Through an Error Message

查詢路徑:

Java\Cx\Java Low Visibility\Information Exposure Through an Error Message 版本:5

類別

PCI DSS v3.2.1: PCI DSS (3.2.1) - 6.5.5 - Improper error handling
OWASP Top 10 2013: A5-Security Misconfiguration
FISMA 2014: Configuration Management
NIST SP 800-53: SI-11 Error Handling (P2)
OWASP Top 10 2017: A6-Security Misconfiguration
OWASP Top 10 API: API3-Excessive Data Exposure
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Error processing
OWASP ASVS: V14 Configuration
OWASP Top 10 2021: A4-Insecure Design
SANS top 25: SANS top 25
ASD STIG 5.2: APSC-DV-002570 - CAT II The application must generate error messages that provide information necessary for corrective actions without revealing information that could be exploited by adversaries.

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=94
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/JwtTokenHelper.java 第 95 行的 validateExpirdToken 方法處理了例外或執行錯誤 e。 在例外處理期間，應用程式在 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/JwtTokenHelper.java validateExpirdToken 方法中將例外詳細資訊 printStackTrace 揭露。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/JwtTokenHelper.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/JwtTokenHelper.java
行	98	102
物件	e	printStackTrace

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/helper/JwtTokenHelper.java

方法 public void validateExpirdToken(String token, ActionInfo actionInfo) throws Exception {

```
....
98.  } catch (ExpiredJwtException e) {
....
102.  e.printStackTrace();
```

Race Condition Format Flaw

查詢路徑:

Java\Cx\Java Low Visibility\Race Condition Format Flaw 版本:5

類別

FISMA 2014: System And Information Integrity
NIST SP 800-53: AC-3 Access Enforcement (P1)
OWASP Top 10 2017: A5-Broken Access Control
CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Time and status
OWASP ASVS: V01 Architecture, Design and Threat Modeling
OWASP Top 10 2021: A4-Insecure Design
ASD STIG 5.2: APSC-DV-001995 - CAT II The application must not be vulnerable to race conditions.

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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=106
狀態	新的
Detection Date	6/26/2024 2:23:46 PM

cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/util/ErrorMessageUtil.java 文件中的 createErrorMessage 方法利用了 format，該方法被其他並行功能以非thread-safe的方式訪問，這可能會導致此資源的資源競爭。

	來源	目的地
檔案	cicd-form-backend-	cicd-form-backend-

	checkmars/src/main/java/com/scsb/cicdform/util/ErrorMessageUtil.java	checkmars/src/main/java/com/scsb/cicdform/util/ErrorMessageUtil.java
行	28	28
物件	format	format

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/util/ErrorMessageUtil.java
方法 public static Map<String, String> createErrorMessage(ErrorMessageEnum errorCode, Object... args) {

```
....
28.  return Collections.singletonMap(errorCode.getErrorCode(),
MessageFormat.format(errorCode.getMessage(), args));
```

Spring Missing Expect CT Header

查詢路徑:
Java\Cx\Java Spring\Spring Missing Expect CT Header 版本:3

類別

CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Verification and representation of input data
OWASP ASVS: V05 Validation, Sanitization and Encoding
OWASP Top 10 2021: A8-Software and Data Integrity Failures
SANS top 25: SANS top 25

描述

Spring Missing Expect CT Header\路徑 1:

嚴重程度：	低風險
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=107
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

Web 應用程式未定義 Expect-CT 標頭，使其更容易受到攻擊。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/aop/RESTResourceAOP.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/aop/RESTResourceAOP.java
行	15	15
物件	annotation	annotation

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/aop/RESTResourceAOP.java
方法 import org.springframework.beans.factory.annotation.Autowired;

```
....
15.  import org.springframework.beans.factory.annotation.Autowired;
```

Spring Missing Content Security Policy

查詢路徑:

Java\Cx\Java Spring\Spring Missing Content Security Policy 版本:3

類別

OWASP ASVS: V14 Configuration
OWASP Top 10 2021: A7-Identification and Authentication Failures

描述

Spring 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://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=108
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

Web 應用程式中未明確定義內容安全性原則。

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/aop/RESTResourceAOP.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/aop/RESTResourceAOP.java
行	15	15
物件	annotation	annotation

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/aop/RESTResourceAOP.java

方法 import org.springframework.beans.factory.annotation.Autowired;

```
....
15.  import org.springframework.beans.factory.annotation.Autowired;
```

Insufficient Logging of Exceptions

查詢路徑:

Java\Cx\Java Best Coding Practice\Insufficient Logging of Exceptions 版本:2

類別

OWASP ASVS: V07 Error Handling and Logging
OWASP Top 10 2021: A9-Security Logging and Monitoring Failures

描述

Insufficient Logging of Exceptions\路徑 1:

嚴重程度：	資訊
結果狀態：	校驗

線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=166
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

In line 21, the security-critical event is not logged properly and, therefore,the error message has some important details omitted.

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/exception/GlobalExceptionHandler.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/exception/GlobalExceptionHandler.java
行	21	21
物件	handleCustomException	handleCustomException

代碼片斷

檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/exception/GlobalExceptionHandler.java

方法 public ResponseEntity<ErrorResponseDto> handleCustomException(EfsException ex) {

```
....
21.  public ResponseEntity<ErrorResponseDto>
handleCustomException(EfsException ex) {
```

Potential Usage of Vulnerable Log4J

查詢路徑:

Java\Cx\Java Best Coding Practice\Potential Usage of Vulnerable Log4J 版本:1

類別

CWE top 25: CWE top 25
MOIS(KISA) Secure Coding 2021: MOIS(KISA) Verification and representation of input data
OWASP ASVS: V12 Files and Resources
OWASP Top 10 2021: A4-Insecure Design
SANS top 25: SANS top 25
ASD STIG 5.2: APSC-DV-002510 - CAT I The application must protect from command injection.

描述

Potential Usage of Vulnerable Log4J\路徑 1:

嚴重程度：	資訊
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=167
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

The Log4j dependency set in cicd-form-backend-checkmars/pom.xml at line 95 contains known vulnerabilities and should not be used.

	來源	目的地
檔案	cicd-form-backend-checkmars/pom.xml	cicd-form-backend-checkmars/pom.xml

行	95	95
物件	dependency	dependency

代碼片斷
檔案名稱 cicd-form-backend-checkmars/pom.xml
方法 <dependency>

```
....
95.  <dependency>
```

Undocumented API

查詢路徑:
Java\Cx\Java Best Coding Practice\Undocumented API 版本:2

類別

OWASP ASVS: V01 Architecture, Design and Threat Modeling

描述

Undocumented API\路徑 1:

嚴重程度：	資訊
結果狀態：	校驗
線上結果	http://10.10.2.164/CxWebClient/ViewerMain.aspx?scanid=85720&projectid=31446&pathid=168
狀態	新的
Detection Date	6/26/2024 2:23:47 PM

The application's uploadFile method (line 45) is an undocumented API endpoint. This is not a best practice and may lead to unintended behavior.

	來源	目的地
檔案	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java	cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
行	45	45
物件	uploadFile	uploadFile

代碼片斷
檔案名稱 cicd-form-backend-checkmars/src/main/java/com/scsb/cicdform/controller/AttachFileController.java
方法 public String uploadFile(@RequestParam("file") MultipartFile file,

```
....
45.  public String uploadFile(@RequestParam("file") MultipartFile file,
```

Reflected XSS All Clients

風險

可能發生什麼問題

成功的跨站腳本 (XSS) 攻擊可讓攻擊者重寫網頁並插入惡意腳本，從而改變原本的輸出。這包括 HTML 片段、CSS 樣式規則、任意 JavaScript 或第三方程式碼的引用。攻擊者可以利用此漏洞來竊取使用者的密碼，收集個人資料，例如信用卡詳細訊息，提供虛假訊息或運行惡意軟件。從受害者的角度來看，這是由正確的網站執行，並且受害者會責怪網站造成的損害。
攻擊者可以使用社交工程學方法引導使用者傳送修改過的輸入，並在要求的網頁中返回該輸入。

原因

如何發生

該應用程式建立的網頁包含不受信任的資料，無論是從使用者輸入、應用程式資料庫或其他外部來源。這些不受信任的資料直接嵌入頁面的 HTML 中，導致瀏覽器將其顯示為網頁的一部分。如果輸入包含 HTML 片段或 JavaScript，這些也會被顯示，而使用者無法判斷這不是預期的網頁。此漏洞是由於直接嵌入任意資料，而未先將其編碼為防止瀏覽器將其視為 HTML 或程式碼而非純文本的格式所導致。
請注意，攻擊者可以透過修改網址或在使用者輸入及其他請求欄位提交惡意資料來利用此漏洞。

一般建議

如何避免

- 在輸出前，必須將所有動態資料完全編碼，不論來源為何。
- 編碼應該是對上下文敏感的。例如：
  - 對於HTML內容使用HTML編碼
  - 對於輸出到屬性值的資料使用HTML屬性編碼
  - 對於伺服器生成的JavaScript使用JavaScript編碼
- 建議使用平台提供的編碼功能，或已知的安全庫來編碼輸出。
- 實施Content Security Policy（CSP），僅為應用程序的資源明確白名單。
- 作為額外的保護層，無論來源如何，驗證所有不受信任的資料（注意，這不是編碼的替代）。驗證應基於白名單：僅接受符合指定結構的資料，而不是拒絕不良模式。檢查：
  - 資料類型
  - 大小
  - 範圍
  - 格式
  - 預期值
- 在Content-Type HTTP回應標頭中，明確定義整個頁面的字符編碼（字符集）。
- 為"深度防禦"，在會話cookie上設置HTTPOnly標誌，以防止任何成功的XSS攻擊竊取cookie。

程式碼範例

Java

Returning Data To Clients Without Encoding

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

    response.setContentType("text/html;charset=UTF-8");

      PrintWriter out = response.getWriter();
      String loc = request.getParameter("location");

      out.println("<h1> Location: " + loc + "<h1>");
}
```

Returning Data to Clients After Encoding The User Input

```
// Using HtmlEscapers by Google Guava

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

    response.setContentType("text/html;charset=UTF-8");

      PrintWriter out = response.getWriter();
      String loc = request.getParameter("location");
      String escapedLocation = HtmlEscapers.htmlEscaper().escape(loc);

      out.println("<h1> Location: " + escapedLocation + "<h1>");
}
```

Absolute Path Traversal

風險

可能發生什麼問題

攻擊者可以任意設定應用程式使用的檔案路徑，這可能導致：

- 竊取敏感文件，例如配置文件或系統文件
- 覆蓋文件，例如程式的二進制(binary)文件、配置文件或系統文件
- 刪除重要文件，形成阻斷服務攻擊(DoS)

原因

如何發生

應用程式在存取本機磁碟上檔案的路徑中，包含了外部輸入的資料。這使攻擊者能夠任意地決定檔案路徑。

一般建議

如何避免

1. 理想情況下，避免依賴動態資料進行檔案選擇。
2. 驗證所有輸入，不管來源如何。驗證應基於白名單：僅接受符合指定結構的資料，而不是拒絕不良模式。檢查：
   - 資料類型
   - 大小
   - 範圍
   - 格式
   - 期望值
3. 僅為檔案名接受動態資料，而不是路徑和資料夾。
4. 確保檔案路徑被完全規範化。
5. 明確限制應用程式僅使用與應用程式二進位資料夾分離的指定資料夾。
6. 限制應用程式的操作系統使用者對必要檔案和資料夾的權限。應用程式不應該能夠寫入應用程式二進位資料夾，也不應該讀取應用程式資料夾和數據資料夾之外的任何內容。

程式碼範例

Input Path Not Canonicalized

風險

可能發生什麼問題

攻擊者可以任意設定應用程式使用的檔案路徑，這可能導致：

- 竊取敏感文件，例如配置文件或系統文件
- 覆蓋文件，例如程式的二進制(binary)文件、配置文件或系統文件
- 刪除重要文件，形成阻斷服務攻擊(DoS)

原因

如何發生

應用程式在存取本機磁碟上檔案的路徑中，包含了外部輸入的資料。這使攻擊者能夠任意地決定檔案路徑。

一般建議

如何避免

1. 理想情況下，避免依賴動態資料進行檔案選擇。
2. 驗證所有輸入，不管來源如何。驗證應基於白名單：僅接受符合指定結構的資料，而不是拒絕不良模式。檢查：
   - 資料類型
   - 大小
   - 範圍
   - 格式
   - 期望值
3. 僅為檔案名接受動態資料，而不是路徑和資料夾。
4. 確保檔案路徑被完全規範化。
5. 明確限制應用程式僅使用與應用程式二進位資料夾分離的指定資料夾。
6. 限制應用程式的操作系統使用者對必要檔案和資料夾的權限。應用程式不應該能夠寫入應用程式二進位資料夾，也不應該讀取應用程式資料夾和數據資料夾之外的任何內容。

程式碼範例

Java

Absolute Path Traversal in "filename" Parameter

```
private String getFileContents(HttpServletRequest request) throws ServletException,
FileNotFoundException, IOException {
      String filename = request.getParameter("filename");
      Path path = Paths.get(filename);
      byte[] fileContentBytes = Files.readAllBytes(path);
      String fileContents = new String(fileContentBytes, FILE_CONTENT_ENCODING_STRING);
      return fileContents;
}
```

Relative Path Traversal in "filename" Parameter

```
private String getFileContents(HttpServletRequest request) throws ServletException,
FileNotFoundException, IOException {
```

```
        String filename = request.getParameter("filename");
        Path path = Paths.get(SERVED_FILES_DIR + filename);
        byte[] fileContentBytes = Files.readAllBytes(path);
        String fileContents = new String(fileContentBytes, FILE_CONTENT_ENCODING_STRING);
        return fileContents;
}
```

Path Traversal Mitigated via Sanitization of Path Variable

```
private static String sanitizePathTraversal(String filename) {
        Path p = Paths.get(filename);
        return p.getFileName().toString();
}

private String getFileContents_fixed(HttpServletRequest request) throws ServletException,
FileNotFoundException, IOException {
        String filename = sanitizePathTraversal(request.getParameter("filename")); // Ensures
access only to files in a given folder, no traversal
        Path path = Paths.get(SERVED_FILES_DIR + filename);
        byte[] fileContentBytes = Files.readAllBytes(path);
        String fileContents = new String(fileContentBytes, FILE_CONTENT_ENCODING_STRING);
        return fileContents;
}
```

Missing HSTS Header

風險

可能發生什麼問題

如果web config沒有設置HSTS表頭、"max-age"(有效期)不足一年，或是沒有設定"includeSubDomains"(包含所有子網域)，可能會使用戶容易遭受中間人攻擊。

原因

如何發生

許多使用者只要在瀏覽器的網址列輸入網域名稱(Domain Name)即可瀏覽網站，並沒有輸入協定(Protocol)的名稱。瀏覽器會自動假設使用者使用的是HTTP協定而不是有加密機制的HTTPS協定。
在第一次發出前往特定網站的請求時，攻擊者可以執行中間人攻擊並將用戶轉址到攻擊者選擇的惡意網站。為了保護用戶避免受到這種事件的影響，可以幫網站加上 HSTS header(HTTP強制安全傳輸技術)，HSTS 會要求用戶的瀏覽器禁止使用不安全的HTTP協定來連接網站。
當支援HSTS功能的瀏覽器訪問了有設置 header 的網站，就不會透過HTTP與該網站通訊。
一旦為特定網站設定了HSTS header，只要還在"max-age"的期間內，HSTS的設定就依然有效，瀏覽器也會被要求阻擋用戶手動覆蓋和接受不受信任的SSL證書。建議"max-age"值(以秒為單位)的設定至少要一年(31536000秒)以上。

一般建議

如何避免

- 在設定 HSTS header 前 - 請先考慮設定後的影響：
  - 強制使用https會阻檔某些需要使用HTTP的狀況，例如功能測試
  - 關閉HSTS並不容易，除了替網站關閉HSTS，客戶的瀏覽器上也要關閉，才能再次使用http連線
- 在應用程式的程式碼中明確設定HSTS header，或是設定在web config檔中。
- 確認HSTS表頭的"max-age"值設置為31536000(含)以上，保證HSTS的有效期至少有一年。
- 一旦使用HSTS表頭並將網頁應用程式的地址提交到HSTS preload list，能夠確保即使使用者是第一次訪問這個網頁應用程式，支援HSTS preload list的瀏覽器會直接將使用者導入到Https協定。但需要注意的是，想要使用HSTS preload list服務，需要有受信任的SSL證書並且設定"max-age"至少1年(31536000秒)的HSTS標頭。
- 注意， 這個弱點判斷程式碼中沒有設定 HSTS header時，會從全部的 response 中，標記第一個案例 作為弱點代表。所以如果只針對指出的案例進行修復，下一次掃描還會再顯示第二個案例，因此建議針對整個應用程式的安全性進行 HSTS header 的部署。在程式碼中設定HSTS時，需要確認整個應用程式都有一併設定完成；若是設定在config檔中則要確保設定是適用於整個應用程式。
- 若IIS版本為IIS 10.0 Version 1709 之前，無法支援在web.config 上設定HSTS Header時，可以參考 https://docs.microsoft.com/en-us/iis/get-started/whats-new-in-iis-10-version -1709/iis-10-version-1709-hsts#solution-2-url-rewrite-module在 IIS 中使用URL Rewrite的方式處理，設定後Checkmarx 8.8.0版會視為已消毒。

程式碼範例

Java

Setting an HSTS Header in an HTTP Response

```
response.setHeader("Strict-Transport-Security", "max-age=31536000; includeSubDomains");
```

XML

Spring - Setting HSTS via Configuration Web.Xml

```
<http>
      <!-- This is a default value -->
      <headers>
            <hsts
                  include-subdomains="true"
                  max-age-seconds="31536000" />
      </headers>
</http>
```

JBoss - Setting HSTS via Configuration Web.Xml

```
<system.webServer>
    <httpProtocol>
        <customHeaders>
            <add name="Strict-Transport-Security" value="max-age=31536000;
includeSubDomains"/>
        </customHeaders>
    </httpProtocol>
</system.webServer>
```

Tomcat - Enable Header Security in Web.Xml

```
<filter>
      <filter-name>httpHeaderSecurity</filter-name>
    <filter-class>org.apache.catalina.filters.HttpHeaderSecurityFilter</filter-class>
      <init-param>
            <param-name>hstsMaxAgeSeconds</param-name>
            <param-value>31536000</param-value>
      </init-param>
      <init-param>
            <param-name>includeSubDomains</param-name>
            <param-value>true</param-value>
      </init-param>
</filter>
```

Cleartext Submission of Sensitive Information

風險

可能發生什麼問題

當敏感且個人的詳細資訊（如密碼、社會安全號碼、信用卡資料以及其他形式的PII（個人身份可識別資訊））在網路上傳輸時，必須始終受到保護。在未加密的通道上發送這些隱私資料，例如未使用SSL/TLS或其他形式的加密，可能會揭露使用者的機密資訊並使其面臨冒充、身份竊取和金融詐欺的風險。

如果SSL/TLS通道在前端網頁伺服器、反向代理或類似伺服器上終止，則可以忽略此問題。

原因

如何發生

應用程式以多種方式處理敏感和隱私資訊。在某一點上，這些機密資料被傳送到網路上，但是應用程式沒有使用SSL/TLS或任何其他安全協議，並且在將其發送到不受保護的通道之前沒有確保資料被加密。

一般建議

如何避免

- 每次透過網路傳輸資料時，皆需保護所有PII和其他敏感資料。
- 在傳輸敏感資料時使用SSL/TLS。或者，還可以使用其他加密協議，如IPsec或SSH。
- 重新考慮應用程式是否需要這些個人詳細資訊。
- 在Web應用程式中，不要在無法確保通道安全的情況下，將個人資料輸出。
- 不要直接將個人資料寫入Standard Socket。取而代之的是，請使用SSLSocket確保通道使用SSL/TLS。

程式碼範例

Java

Using Standard Socket (No Encryption) for a Basic Authentication Server

```
public void runServer() {
    ServerSocket server = new ServerSocket(PORT);
    Socket client;

    while (true) {
        client = server.accept();
        MyUser user = handleRequest(client.getInputStream());

        PrintWriter output = new PrintWriter(server.getOutputStream());
        output.println(user.AccountId);
        output.flush();
    }
}
```

Using Encrypted SSLSocket for a Basic Authentication Server

```
public void runServer() {
    try {
        SSLServerSocketFactory factory =
            (SSLServerSocketFactory) SSLServerSocketFactory.getDefault();
```

```
        ServerSocket server = factory.createServerSocket(PORT);
        Socket client;

        while (true) {
            client = server.accept();
            MyUser user = handleRequest(client.getInputStream());

            PrintWriter output = new PrintWriter(server.getOutputStream());
            output.println(user.AccountId);
            output.flush();
        }
    }
    catch (IOException ex) {
        handleException(ex);
    }
    finally {
        if (output != null) output.close();
        if (client != null)
            if (!client.isClosed()) client.close();
        if (server != null)
            if (!server.isClosed()) server.close();
    }
}
```

Authenticating Server to Server via HTTP

```
public static boolean authenticateServerToServer(String username, String password) throws
IOException {
    String stringUrl = "http://" + HOSTNAME + "/" + URI_PATH;
    URL url = new URL(stringUrl);
    HttpURLConnection conn = (HttpURLConnection)url.openConnection();
    conn.setRequestMethod("POST");
    conn.setDoOutput(true);

    String postParameters  = "username=" + username + "&password=" + password;
    byte[] postRequestBytes = postParameters.getBytes();
    OutputStream os = conn.getOutputStream();
    os.write(postRequestBytes);
    os.flush();
    os.close();
    return conn.getResponseCode() == HttpURLConnection.HTTP_OK;
}
```

Authenticating Server to Server via HTTPS

```
public static boolean authenticateServerToServer(String username, String password)  throws
IOException {
    String stringUrl = "https://" + HOSTNAME + "/" + URI_PATH;
    URL url = new URL(stringUrl);
    HttpsURLConnection conn = (HttpsURLConnection)url.openConnection();
    conn.setRequestMethod("POST");
    conn.setDoOutput(true);

    String postParameters  = "username=" + username + "&password=" + password;
    byte[] postRequestBytes = postParameters.getBytes();
    OutputStream os = conn.getOutputStream();
    os.write(postRequestBytes);
    os.flush();
    os.close();
    return conn.getResponseCode() == HttpURLConnection.HTTP_OK;
```

```
}
```

SSRF

風險

可能發生什麼問題

攻擊者可以利用此漏洞發出任何來源為應用伺服器的請求。這可以被利用來掃描內部服務、代理攻擊受保護的網路、繞過網路控制、下載未經授權的文件、訪問內部服務及管理介面以及可能控制請求內容甚至竊取伺服器憑證。

原因

如何發生

應用程式接收從使用者端傳來的URL，然後將此當作請求傳送給另一個遠端伺服器。
然而，攻擊者可以在請求中注入任意的URL，造成應用程式連線至任意一個攻擊者想要的伺服器。所以，攻擊者可以濫用該應用程式來訪問本來無法訪問的服務，而表面上這個請求來自應用伺服器。

一般建議

如何避免

- 不直接利用使用者輸入對任意服務進行連線。
- 如果可以，應用程式應該讓使用者的瀏覽器直接檢索所需的資訊。
- 如果應用程式需要在伺服器上代理請求，明確地將允許的URL列入白名單，並且不包括任何敏感的伺服器資訊。

程式碼範例

Java

Retrieve and Display Contents of URL

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
  if (request.getParameterMap().containsKey("url")) {
    String url = request.getParameter("url");
    PrintWriter out = response.getWriter();
    URL u = new URL(url);
    InputStreamReader sr = new InputStreamReader(u.openConnection().getInputStream());
    BufferedReader reader = new BufferedReader(sr);
    String line = reader.readLine();
    while (line != null) {
      out.write(line);
      line = reader.readLine();
    }
  }
}
```

Validate and Redirect User's Browser

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
  if (request.getParameterMap().containsKey("url")) {
    String url = request.getParameter("url");
    if (url.startsWith("/") && !url.startsWith("//")) {
      response.sendRedirect(url);
    } else {
```

```
        response.sendRedirect("/");
      }
    }
  }
```

Stored Absolute Path Traversal

風險

可能發生什麼問題

攻擊者可以定義應用程式使用的任意檔案路徑，可能導致：

- 竊取敏感檔案，如配置或系統檔案
- 覆寫檔案，例如程式二進位檔、config檔或系統檔案
- 刪除關鍵檔案，導致拒絕服務（DoS）

原因

如何發生

應用程式透過儲存的資料來確定應用程式伺服器上訪問檔案的檔名。如果這些資料可能透過使用者輸入污染，致使攻擊者可以儲存任意值，以執行路徑遍歷。

一般建議

如何避免

1. 理想情況下，避免依賴動態資料進行檔案選擇。
2. 驗證所有輸入，不管來源如何。驗證應基於白名單：僅接受符合指定結構的資料，而不是拒絕不良模式。檢查：
    - 資料類型
    - 大小
    - 範圍
    - 格式
    - 期望值
3. 僅為檔案名接受動態資料，而不是路徑和資料夾。
4. 確保檔案路徑被完全規範化。
5. 明確限制應用程式僅使用與應用程式二進位資料夾分離的指定資料夾。
6. 限制應用程式的操作系統使用者對必要檔案和資料夾的權限。應用程式不應該能夠寫入應用程式二進位資料夾，也不應該讀取應用程式資料夾和數據資料夾之外的任何內容。

程式碼範例

Java

Stored Path Traversal from DB

```
String query = "SELECT username FROM users WHERE id = ?";
PreparedStatement pstmt = con.prepareStatement(query);
pstmt.setInt(userID, 1);

ResultSet resultSet = pstmt.executeQuery();
resultSet.next();
path = resultSet.getString(1);
con.close();

File file = new File(path);
Scanner scanner = new Scanner(file);
System.out.println(scanner.nextLine());
scanner.close();
```

Stored Path Traversal from DB Mitigated by Utilizing Paths.get() and getName()

```
String query = "SELECT username FROM users WHERE id = ?";
PreparedStatement pstmt = con.prepareStatement(query);
pstmt.setInt(userID, 1);

ResultSet resultSet = pstmt.executeQuery();
resultSet.next();
path = resultSet.getString(1);
con.close();

File file = new File(path);
String fileName = file.getName();
Path safePath = Paths.get("userFiles", fileName);
file = new File(safePath.toString());

Scanner scanner = new Scanner(file);
System.out.println(scanner.nextLine());
scanner.close();
```

Unsafe Object Binding

風險

可能發生什麼問題

不安全地將物件綁定到request可能會使未預期的設定器暴露給遠程攻擊者，使其能夠通過將設定器廣泛綁定到傳入的request來直接訪問物件、屬性甚至物件中的物件。

原因

如何發生

使用內建於MVC控制器中的物件綁定方法，會將所有公開的設定器暴露出來，以便輕鬆地將使用者在表單中提交的值與它們所要創建或修改的物件和屬性進行連接。這種方法使應用程式能夠跳過為了解析使用者輸入值而必須逐個手動設定的樣板程式碼。

然而，這也可能對應用程式的邏輯和流程構成重大風險 - 當以這種方式盲目地批量綁定物件時，也可能意外地暴露出未預期的物件或屬性，從而被攻擊者篡改。

一般建議

如何避免

- 審查所有大量指定的物件，以確保這種方法不會無意間暴露出意外的公開設定器或建構函式。
- 確保必要時，應用程式程式碼正確限制對屬性和物件的訪問權限。
- 考慮從物件綁定方法轉向更細緻的方法，只有明確設定值，以防止意外地將未預期的值暴露出來並受到暗中修改。

程式碼範例

Java

Unsafe Object Binding via Spring ModelAttribute

```
//Item Bean:
public class Item {
      private String id;
    private String itemName;
    private String price;
    private String shippingAddress;
    private User buyer;
      //Public setters/getters:
    [..]
}
//User Bean:
public class User {
      private String id;
    private String userName;
    private String password;
      //Public setters/getters:
    [..]
}

//Item Controller:
@Controller
public class ItemController {
      @RequestMapping(value="saveItem", method = RequestMethod.POST)
      public String saveItem(@ModelAttribute("item") Item item, ModelMap model) {
            db.save(item); //If the parameter "user.password=hacked!!" is added, the password
for the user is changed to "hacked!!".
```

```
            return "saveItemView";
        }
    }
```

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

Java

Loading a Library without Catching

```
public static void loadLib() {
      System.loadLibrary(LIB_NAME); // If LIB_NAME does not exist, an unhandled exception will
be thrown
}
```

Handle All Possible Exceptions within the Error-Prone Method

```
public static void loadLib() {
      try {
            System.loadLibrary(LIB_NAME);
    } catch (SecurityException se) {
                  // Handle SecurityException
      } catch (UnsatisfiedLinkError sle) {
                  // Handle UnsatisfiedLinkError
      } catch (NullPointerException npe) {
                  // Handle NullPointerException
      }
}
```

Aggregate Potential Exceptions to Calling Code

```
public static void loadLib() throws UnsatisfiedLinkError, NullPointerException,
SecurityException {
      System.loadLibrary(LIB_NAME);
}
```

Improper Resource Access Authorization

風險

可能發生什麼問題

未經授權的操作可能允許攻擊者將惡意內容或程式碼寫入檔案、資料庫和其他輸入/輸出，或讀取敏感的輸入/輸出內容。此問題的影響因實作方式而異，但可能造成以下情況：

- 遠端程式碼執行：若攻擊者能夠將惡意資料注入可寫入的輸入/輸出，該資料可能會被編譯為程式碼，從而執行惡意程式碼
- 覆寫或洩漏設定檔案
- 危害儲存數據的機密性或完整性

原因

如何發生

程式碼中的邏輯流程觸發了I/O 操作，但未經授權。如果攻擊者能夠觸發它，則容易遭受攻擊。

一般建議

如何避免

確認邏輯流程是會受到使用者輸入或行為的影響時，請始終確保使用者有授權觸發它們。

程式碼範例

Java

Writing to File Without Any Authorization Checks

```
Part filePart = request.getPart("file");
if (filePart != null) {
      InputStream filecontent = null;
      filecontent = filePart.getInputStream();
      Path path = Paths.get(filename);
      byte[] contentByteArray = new byte[filecontent.available()];
      filecontent.read(contentByteArray);
      Files.write(path, contentByteArray);
}
```

Using a Basic Authorization Check Based on Session Variables

```
HttpSession session = request.getSession();
String role = (String)session.getAttribute("role");
if (role.equals(ADMIN)) {
      Part filePart = request.getPart("file");
      if (filePart != null) {
            InputStream filecontent = null;
            filecontent = filePart.getInputStream();
            Path path = Paths.get(filename);
            byte[] contentByteArray = new byte[filecontent.available()];
            filecontent.read(contentByteArray);
            Files.write(path, contentByteArray);
      }
```

```
}
```

Improper Resource Shutdown or Release

風險

可能發生什麼問題

未釋放的資源會造成可供系統使用的資源耗盡，最終導致可靠性和可用性的問題，例如效能退化、程序膨脹和系統不穩定。如果攻擊者有意洩漏此資訊，則可能引起廣泛的 拒絕服務(Denial of Service,DoS) 攻擊。如果資源在後續配置之間繼續保留資料或使用者ID，就可能會在未授權的使用者之間公開敏感資訊。

原因

如何發生

應用程式分配資源，卻無法保證這些物件都會被及時的關閉和釋放，這其中可能包括資料庫連線、資料處理、網路插座(network socket)或其他任何需要被釋放的資源。系統在正常執行的情況下這些資源都要被正確釋放；但如果執行中出現任何異常，就可能導致這些未被正確關閉或釋放的資源被洩漏。

需要注意的是，即使在像 Java 語言的記憶體管理中，這些資源也需要被明確釋放。而許多類型的資源即使不再使用，在垃圾回收器(Garbage Collector)中也不一定馬上會被回收；即使物件最終會釋放資源，我們也無法控制垃圾回收器何時執行。

一般建議

如何避免

- 一律關閉和釋放所有資源
- 確保在`finally { }`區塊中釋放資源(以及其他必要的清理工作)。不要在`catch { }`區塊中關閉資源，因為這不保證會被呼叫。
- 在實作`Closable`或`AutoClosable`介面的類別時，明確地呼叫`.close()`。
- 更好的解決方案是使用 try-with-resources，以便自動關閉任何已設定的`AutoClosable`介面。

程式碼範例

Java

Unreleased Database Connection

```
private MyObject getDataFromDb(int id)  {
      MyObject data = null;
      Connection con = null;
      try {
            Connection con = DriverManager.getConnection(CONN_STRING);
            data = queryDb(con, id);
      }
      catch ( SQLException e ) {
            handleError(e);
      }
}
```

Explicit Release of Database Connection

```
private MyObject getDataFromDb(int id)  {
      MyObject data = null;
      Connection con = null;
```

```
      try {
            Connection con = DriverManager.getConnection(CONN_STRING);
            data = queryDb(con, id);
      }
      catch ( SQLException e ) {
            handleError(e);
      }
      finally {
            if ((con != null) && (!con.isClosed())) {
            con.close();
         }
     }
}
```

Automatic Implicit Release Using Try-With-Resources

```
private MyObject getDataFromDb(int id)  {
      MyObject data = null;
      Connection con = null;
      try (Connection con = DriverManager.getConnection(CONN_STRING)) {
            data = queryDb(con, id);
      }
      catch ( SQLException e ) {
            handleError(e);
      }
}
```

Information Exposure Through an Error Message

風險

可能發生什麼問題

透露關於程式的環境、使用者或相關資訊 (例如：stack trace) 將會讓攻擊者找到其他的缺失也幫助攻擊者來發起攻擊。這也可能會使機密資料洩露，例如：密碼或資料庫欄位。

原因

如何發生

應用程式以不安全的方式處理例外(exception)，包括直接在 error message 中顯示完整的原始詳細訊息。這在幾個狀況下都可能會發生：不處理 exception；直接將 exception 輸出到頁面或檔案中；顯式return exception 物件；設定檔設定不嚴謹。這些 exception 細節可能包含機密資訊，並隨著 Runtime Error 而流出。

一般建議

如何避免

- 不將 excecption 資訊直接輸出或是透露給使用者，建議回傳一個制式化的錯誤訊息。Exception細節則應記錄於Log機制內。
- 任何會拋出 exception 的函式都應該要被包在處理 exception 的區塊內，而處理的方式有:
  - 明確的處理預期內的exception。
  - 包含了一個預設的解決方式來處理無預期的exceptions。
- 設定一個全域處理器來避免無處理的錯誤被送至使用者端。

程式碼範例

Java

Handle Exception by Printing To Output

```
private void wrapCallToDB_Unsafe(HttpServletRequest request)
            throws ServletException, IOException {
      String paramValue = request.getParameter("Param");

      try {
            callDbProc(paramValue);
      } catch (SQLException ex) {
            ex.printStackTrace();
      }
}
```

Write Exception Details to Log, Send Generic Error Message

```
private void wrapCallToDB_SafePrintToLog(HttpServletRequest request)
            throws ServletException, IOException {
      String paramValue = request.getParameter("Param");

      try {
```

```
            callDbProc(paramValue);
        } catch (SQLException ex) {
            writeExceptionToLog(ex);
            System.err.println("Database Error, see log for details");
        }
    }
```

Incorrect Permission Assignment For Critical Resources

風險

可能發生什麼問題

具有危險權限的檔案可能允許攻擊者從這些檔案的內容中擷取敏感資訊、篡改其內容或潛在地執行它們。

原因

如何發生

檔案或目錄被創建時具有危險的權限，可能是有設置這些權限，不然就是依賴於不安全的默認權限。

一般建議

如何避免

- 請始終明確設定檔案的權限
- 請勿將危險權限設定在檔案上
    - 當決定使用者可以讀取、寫入或執行哪些檔案時，請始終考慮最小權限原則，並且只在必要時授予這些權限

程式碼範例

Java

Writing A File with Implicit Permissions

```
File tempFile = File.createTempFile(TEMP_FILE_PREFIX,TEMP_FILE_SUFFIX, new
File(TEMP_FOLDER));
FileWriter fw = new FileWriter(tempFile);
fw.write(CONTENT);
```

Writing A File with Explicit Permissions

```
File tempFile = File.createTempFile(TEMP_FILE_PREFIX,TEMP_FILE_SUFFIX, new
File(TEMP_FOLDER));
FileWriter fw = new FileWriter(tempFile);
tempFile.setExecutable(false);
tempFile.setReadable(true);
tempFile.setWritable(true);
fw.write(CONTENT);
```

Log Forging

風險

可能發生什麼問題

攻擊者可以偽造安全敏感動作的稽核記錄，並留下虛假的稽核軌跡，可能會牽連無辜的使用者或隱藏事件。

原因

如何發生

當應用程式執行安全與敏感性的動作時，會記錄稽核日誌(audit log)。由於稽核日誌中包含使用者輸入的內容，這些內容既沒有經過驗證資料類型，也沒有經過適當的處理，因此該輸入可能包含假資訊，假裝成合法的稽核日誌資料。

一般建議

如何避免

1. 不管來源為何所有輸入都要進行驗證，且驗證要根據白名單：只接受符合指定格式的資料，而不是使用黑名單。請檢查：
    - 資料型態(Data type)
    - 資料大小(Size)
    - 資料範圍(Range)
    - 資料格式(Format)
2. 驗證不能取代 encode。不論來源為何要將所有的外部資料進行 encode 後才能寫進 Log。
3. 使用安全的 Logging 機制。

程式碼範例

Java

User Input Affects Logging

```
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

    String color = request.getParameter("color");
    logger.info("{} was picked", color);
      if colorList.contains(color){
      // Handle Response
    }else{
      // Handle Response
    }
}
```

User Input Encoded Prior Logging

```
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

    String color = request.getParameter("color");
      cleanColor = color.replace('\t', '_').replace('\n', '_').replace('\r', '_');
    logger.info("{} was picked", cleanColor);
```

```
      if colorList.contains(cleanColor){
       // Handle Response
     }else{
       // Handle Response
     }
 }
```

Race Condition Format Flaw

風險

可能發生什麼問題

資源競爭可能導致錯誤，無效的值或意外行為可能導致拒絕服務。最糟糕的情況是，它可能允許攻擊者通過重新執行可控制的資源競爭來檢索資料或繞過安全管控，直到它對其有利為止。

原因

如何發生

當多個並行邏輯處理使用一個公共的、單一的資源實例時，就會發生資源競爭。如果這些邏輯處理試圖在沒有及時管理系統 (如鎖) 的情況下檢索和更新資源，則會發生資源競爭。
發生資源競爭的一個例子是，可能會將某個值返回流程以進行進一步編輯的資源，然後由第二個更新，從而導致原始流程的資料不再有效。一旦原始流程將不正確的值編輯並更新回資源中，第二個流程的更新將被覆蓋並丟失。

一般建議

如何避免

當跨應用程式在並行流程之間共享資源時，確保這些資源是thread-safe的，或者實現鎖定機制以確保預期的並行活動。

程式碼範例

Java

Testing Static Int Concurrency - Different Threads Increment and Decrement The Same Counter Repeatedly, Resulting in a Race Condition

```
	public static int counter = 0;
	public static void start() throws InterruptedException {
		incrementCounter ic;
		decrementCounter dc;
		while(counter == 0) {
			counter = 0;
			ic = new incrementCounter();
			dc = new decrementCounter();
			ic.start();
			dc.start();
			ic.join();
			dc.join();
		}
		System.out.println(counter); //Will stop and return either -1 or 1 due to race condition over counter
	}

	public static class incrementCounter extends Thread {
	    public void run() {
		counter++;
	    }
	}

	public static class decrementCounter extends Thread {
	    public void run() {
		counter--;
	    }
	}
```

Different Threads Increment and Decrement The Same Thread-Safe Counter Repeatedly, Never Resulting in a Race Condition

```
	public static int counter = 0;
	public static Object lock = new Object();

	public static void start() throws InterruptedException {
		incrementCounter ic;
		decrementCounter dc;
		while(counter == 0) { // because of proper locking, this condition is never false
			counter = 0;
			ic = new incrementCounter();
			dc = new decrementCounter();
			ic.start();
			dc.start();
			ic.join();
			dc.join();
		}
		System.out.println(counter); // Never reached
	}

	public static class incrementCounter extends Thread {
	    public void run() {
		synchronized (lock) {
			counter++;
		}
	    }
	}

	public static class decrementCounter extends Thread {
	    public void run() {
		synchronized (lock) {
			counter--;
		}
	    }
	}
```

Testing Formatter Race Condition - Java Formatters are Not Thread Safe

```
	static Locale defaultLocale = new Locale("en", "US", "USD");
	static NumberFormat numberFormatter = NumberFormat.getCurrencyInstance(defaultLocale);

	public static class CurrencyFormatRunnable implements Runnable {

		double money;
		public CurrencyFormatRunnable(Object parameter) {
		    this.money = (double)parameter;
		}

		// Format a double as a USD string; if formatter result does not match a known value (100.0 -> $100.00, 50.0 -> $50.00),
		// as expected with a Race Condition, a warning is printed and the application closes.
		public void run() {
			String formattedMoney;
			formattedMoney = numberFormatter.format(money);
			if (!formattedMoney.equals("$"+money+"0")) {
				System.out.println("Formatted number was $" + money + ", but the result was " + formattedMoney +"!");
				System.exit(1);
		/*	Potential outputs:
	*	Formatted number was $50.0, but the result was $100.00!
```

```
            *       Formatted number was $50.0, but the result was 100.00!
                                    *       Formatted number was $100.0, but the result was $50.00!
            *       Formatted number was $100.0, but the result was $$50.00!
            *       Note the erroneous behavior with completely incorrect values, like $$50.00, which
occurs
            *       because the string is read as a new value is written
            */

                        }
                    }
                }

      public static void start() throws InterruptedException {
            Runnable m1;
            Runnable m2;
            Thread t1;
            Thread t2;
            while(true) {
                  m1 = new CurrencyFormatRunnable((double)100.00);
                  m2 = new CurrencyFormatRunnable((double)50.00);
                  t1 = new Thread(m1);
                  t2 = new Thread(m2);
                  // Attempt to concurrently use the formatter
                  t1.start();
                  t2.start();

                  t1.join();
                  t2.join();
            }
      }
```

Formatter Race Condition Mitigated, with Each Thread Using Its Own Instance of Java Formatter

```
      public static class CurrencyFormatRunnable implements Runnable {
                double money;
                public CurrencyFormatRunnable(Object parameter) {
                    this.money = (double)parameter;
                }

                public void run() {
                      String formattedMoney;
                    Locale defaultLocale = new Locale("en", "US", "USD");
                    NumberFormat numberFormatter =
NumberFormat.getCurrencyInstance(defaultLocale);
                      formattedMoney = numberFormatter.format(money);
                      if (!formattedMoney.equals("$"+money+"0")) {
                                  System.out.println("Formatted number was $" + money + ", but
the result was " + formattedMoney +"!");
                      // This is never reached
                                  System.exit(1);
                      }
                }
            }

      public static void start() throws InterruptedException {
            Runnable m1;
            Runnable m2;
            Thread t1;
            Thread t2;
            while(true) {
                  m1 = new CurrencyFormatRunnable((double)100.00);
                  m2 = new CurrencyFormatRunnable((double)50.00);
                  t1 = new Thread(m1);
                  t2 = new Thread(m2);
                  // Attempt to concurrently use the formatter
```

```
                t1.start();
                t2.start();

                t1.join();
                t2.join();
            }
        }
```

Spring Missing Expect CT Header

風險

可能發生什麼問題

如果未能設置 Expect-CT 標頭並向其提供 "enforce" 參數和至少為一年的合理 "max-age" 值，可能會讓用戶容易受到"中間人"攻擊。

原因

如何發生

使用 SSL/TLS 時，瀏覽器會根據客戶端的已識別 CA（證書頒發機構）列表來驗證伺服器在連接握手期間發送的證書。該模型的安全功能將信任從伺服器轉移到 CA。
聲明 Expect-CT 標頭將使受支持的瀏覽器使用證書透明度來檢測 CA 完整性是否受到危害，並根據標頭參數中的定義來報告和/或強制執行安全連接。
使用帶有 Expect-CT 的證書透明度和正確參數，可避免中間人攻擊。

一般建議

如何避免

- 在設置 Expect-CT 標頭之前 - 考慮其可能產生的影響：
  - 強制執行 Expect-CT 將阻止未來使用 HTTP，這可能會阻礙某些測試
  - 禁用 Expect-CT 並非易事，因為一旦在網站上將其禁用，也必須在瀏覽器上將其禁用
- 通過設置沒有 'enforce' 標誌的 Expect-CT 標頭來測試您的環境以檢查是否存在證書問題 - 然後使用 'enforce' 標誌
- 在應用程式程式碼中顯式設置 Expect-CT 標頭，或使用 Web 伺服器配置。
- 確保 Expect-CT 標頭的 "max-age" 值設置為 31536000 以確保證書透明度的使用嚴格執行至少一年。
- 如果通過配置添加此標頭，請確保此配置適用於整個應用程式。

默認情況下，Spring Security 不會添加此標頭。

程式碼範例

XML

Adding Expect_CT Header Using Spring Security's XML Configuration

```
<http>
        <headers>
               <header name="Expect-CT" value="max-age=3600, enforce"/>
        </headers>
</http>
```

Java

Adding Expect_CT Header Using Spring Security's Java Configuration

```
@EnableWebSecurity
```

```
public class WebSecurityConfig extends
WebSecurityConfigurerAdapter {

@Override
protected void configure(HttpSecurity http) throws Exception {
      http
        // ...
        .headers()
             .addHeaderWriter(new StaticHeadersWriter("Expect-CT","max-age=3600, enforce"));
}
}
```

Spring Missing Content Security Policy

風險

可能發生什麼問題

Content-Security-Policy header 強制內容的來源，例如腳本的來源、嵌入（子）框架、嵌入（父）框架或圖像，被當前網頁信任和允許；如果在網頁中，內容的來源不遵守嚴格的內容安全性原則，瀏覽器會立即拒絕該內容。 未定義策略可能會使應用程式的使用者暴露於跨網站腳本 (XSS) 攻擊、點擊劫持攻擊、內容偽造等。

原因

如何發生

Content-Security-Policy header被現代瀏覽器用作可信內容來源的指示器，包括媒體、圖像、腳本、框架等。 如果未明確定義這些策略，則默認瀏覽器行為將允許不受信任的內容。

應用程式創建 Web response，但未正確設置 Content-Security-Policy 標頭。

一般建議

如何避免

建議根據業務需求和外部檔案託管服務的部署，明確地設定合適的CSP標頭(框架、腳本、表單、腳本、媒體、圖片等...)，具體來說，不要使用萬用字元"*"來指定這些策略，因為這將允許來自外部的任何資源內容。

CSP可以在網頁應用程序代碼中明確定義，作為由web-server 配置所管理的標頭，或在HTML<head>下的 <meta>標籤中定義。

程式碼範例

Java

Adding CSP Header Using Spring Security Java Configuration

```
@Configuration
public class SpringSecurityConfig extends WebSecurityConfigurerAdapter {

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        // Add CSP headers
        http.headers()
            .contentSecurityPolicy("script-src 'self' https://example.com; object-src
https://example.com; report-uri /csp-report-endpoint/");
    }
}
```

XML

Adding CSP Header Using Spring Security XML Configuration

```
<http>
   <!-- ... -->

    <headers>
```

```
    <content-security-policy policy-directives="script-src 'self' https://apis.example.com">
        </content-security-policy>
    </headers>
</http>
```

Spring Overly Permissive Cross Origin Resource Sharing Policy

風險

可能發生什麼問題

過於寬鬆的跨域資源共享 (CORS) 標頭 "Access-Control-Allow-Origin" 可能會使其他網站的腳本可以訪問、甚至篡改受影響的 web 應用程式上的資源。這些資源包括頁面內容、Token 等，因此可能受到跨站點請求偽造 (CSRF) 或跨站點腳本 (XSS) 攻擊、假冒用戶執行操作，如更改密碼或違反用戶隱私。

原因

如何發生

程式碼中的 Access-Control-Allow-Origin 被錯誤地設置為不安全的值。

默認情況下，現代瀏覽器會根據同源策略 (SOP) 禁止不同域之間的資源共享訪問彼此的 DOM 內容、cookie jar 和其他資源，這是為了避免惡意 Web 應用程序攻擊合法的 Web 應用程式及其用戶。例如——網站 A 默認無法檢索網站 B 的內容，因為這違反了 SOP。使用具體標頭定義的跨域資源共享 (CORS) 策略可以放鬆這個嚴格的默認行為，允許跨站點通信。但是，如果使用不當，CORS 可能會允許過度地廣泛信任 Web 應用程式，使其能夠提交請求並獲得 Web 應用程式的回應，從而執行意外的或潛在惡意的行為。

一般建議

如何避免

如果沒有顯示需求，請不要設置任何 CORS 標頭。如果有需要，請考慮設置這些標頭的業務需求，然後選擇最嚴格的配置，例如可信任的白名單、安全和允許的域訪問，同時使用其他 CORS 標頭嚴格地提供所需的和預期的功能。

Spring Security 具有內置機制，可使用 @CrossOrigin 註釋來配置 CORS 標頭。

Spring 的默認允許來源過於寬鬆，建議手動指定允許的來源。

程式碼範例

Java

Default 'origins' Parameter Allowing All Origins in a Specific Endpoint

```
@RestController
@RequestMapping("/resource")
public class ResourceController {

      @CrossOrigin
      @GetMapping("/{id}")
      public Resource retrieve(@PathVariable Long id) {
            //  ...
      }
}
```

Setting an 'origins' Parameter on a Specific Controller

```
@CrossOrigin(origins = "https://example.com", maxAge = 3600)
@RestController
```

```
@RequestMapping("/resource")
public class ResourceController {

	@GetMapping("/{id}")
	public Resource retrieve(@PathVariable Long id) {
		//  ...
	}
}
```

Applying the CORS Header to Every Endpoint Using Spring Security's Java Configuration

```
@EnableWebSecurity
public class WebSecurityConfig extends WebSecurityConfigurerAdapter {

	@Override
	protected void configure(HttpSecurity http) throws Exception {
		http
			// by default uses a Bean by the name of corsConfigurationSource
			.cors();
	}

	@Bean
	CorsConfigurationSource corsConfigurationSource() {
		CorsConfiguration configuration = new CorsConfiguration();
		configuration.setAllowedOrigins(Arrays.asList("https://example.com"));
		configuration.setAllowedMethods(Arrays.asList("GET","POST"));
		UrlBasedCorsConfigurationSource source = new UrlBasedCorsConfigurationSource();
	  // Applying the CORS to all endpoints
	  source.registerCorsConfiguration("/**", configuration);
		return source;
	}
}
```

XML

Applying Spring Security's Default CORS with an Overly Permissive Configuration

```
<http>
	<cors />
</http>
```

Insufficient Logging of Exceptions

風險

可能發生什麼問題

If security-critical information is not recorded, there will be no trail for forensic analysis and discovering the cause of problems or the source of attacks may become more difficult or impossible.

原因

如何發生

Error's stack trace is not fully printed.

一般建議

如何避免

Use a centralized logging mechanism that supports multiple levels of detail.Be sure to set the level of logging appropriately in a production environment. Sufficient data should be logged to enable system administrators to detect attacks, diagnose errors, and recover from attacks.

程式碼範例

Java

Print Stack Trace

```
public class Example1{
	public String login(Model model, String username, String password) {
		try {
			// attempt to login user
			userService.login(username, password);
		} catch (Exception ex) {
			ex.printStackTrace();
		}
		return "login";
	}
}
```

Log Error

```
public class Example{
	public String login(Model model, String username, String password) {
		try {
			// attempt to login user
			userService.login(username, password);
		} catch (Exception ex) {
			log.error("Exception looking up customer by name: " + ex.getMessage());
		}
		return "login";
	}
}
```

Log Error

```
public class Example{
      public String login(Model model, String username, String password) {
            try {
                  userService.login(username, password);
            } catch (Exception ex) {
                  log.error("Exception looking up customer by name: " + ex.getMessage(),
ex);
            }
            return "login";
      }
}
```

Potential Usage of Vulnerable Log4J

風險

可能發生什麼問題

Using a vulnerable Log4j version might put the application at risk, as multiple CVEs were identified that allow from Denial of Service to Remote Code Execution.

原因

如何發生

The presence of the identified Log4j library version, under specific conditions, could allow an attacker to remotely execute code on the target application.

一般建議

如何避免

To address any known security issues Log4j should be updated to the latest available version. Additional and updated details in: https://logging.apache.org/log4j/2.x/security.html

---

程式碼範例

Java

Setting Log4j 2.3.0 dependency with Maven

```
<dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-core</artifactId>
    <version>2.3.0</version>
</dependency>
```

Setting Log4j 2.3.0 dependency with Gradle

```
implementation group: 'org.apache.logging.log4j', name: 'log4j-core', version: '2.3.0'

implementation 'org.apache.logging.log4j:log4j-core:2.3.0'
```

Undocumented API

風險

可能發生什麼問題

Undocumented or outdated documentation on API endpoints make it more difficult to find and/or fix vulnerabilities.

原因

如何發生

No automatic documentation is being used or if the documentation is being manually specified, some methods are not being correctly documented.

一般建議

如何避免

Generate documentation automatically by adopting open standards. Include the documentation build in your CI/CD pipeline.Document all aspects of your API such as authentication, errors, redirects, rate limiting, cross-origin resource sharing (CORS) policy and endpoints, including their parameters, requests, and responses.

程式碼範例

Java

API Endpoint correctly documented in separate file

```
@Controller
@RequestMapping("/pet")
public class DeferredResultController {

      @GetMapping("/id")
      public @ResponseBody Pet getPetNameById(int id) {
            return context.getPetNameById(id);
    }
}

// OpenAPI specification (openapi.json)
{
  "openapi" : "3.0.1",
  "paths" : {
    "/pet/id" : {
      "get" : {
        "summary" : "Get an existing pet",
        "requestBody" : {
          "description" : "Pet object that needs to be added to the store",
        },
        "responses" : {
          "400" : {
            "description" : "Invalid ID supplied",
            "content" : { }
          }
        }
      }
    }
  }
}
```

API Endpoint not documented

```
@Controller
@RequestMapping("/pet")
public class DeferredResultController {

    @GetMapping("/id")
      public @ResponseBody Pet getPetNameById(int id) {
             return context.getPetNameById(id);
    }

    // This method is not listed on the OpenApi specification file (see below)
    @DeleteMapping("/id")
      public @ResponseBody String deletePetNameById(int id) {
             return context.deletePetNameById(id);
    }
}

// OpenAPI specification (openapi.json)
{
  "openapi" : "3.0.1",
  "paths" : {
    "/pet/id" : {
      "get" : {
        "summary" : "Get an existing pet",
        "requestBody" : {
          "description" : "Pet object that needs to be added to the store",
        },
        "responses" : {
          "400" : {
            "description" : "Invalid ID supplied",
            "content" : { }
          }
        }
      }
    }
  }
}
```

Documentation being automatically generated

```
@Configuration
@EnableSwagger2
public class SpringFoxConfig {
    @Bean
    public Docket api() {
        return new Docket(DocumentationType.SWAGGER_2)
          .select()
          .apis(RequestHandlerSelectors.any())
          .paths(PathSelectors.any())
          .build();
    }
}
```

檢測的語言

語言	HASH值	變更的日期
Java	5683964027843638	2024/6/11
PLSQL	0342189457118079	2024/6/11
Common	1330881790325397	2024/6/11