Sparrow Analysis Report

Sparrow Cloud

Analysis Summary

Project Name	Demo
Analysis ID	723
Analysis Type	File
Started at	2025-06-11 10:07:09
Ended at	2025-06-11 10:13:52
Analyzing Time	6 m 42 s
Total Issues	273
Printed Issues	273

Risk Level and Issues

Critical	High	Medium	Low	Trivial
19	94	157	3	0

Reference and Issues

Reference Name	Total Issues
.NET framework design guideline	0
CWE 658 4.14	0
CWE 658 4.7	0
CWE 659 4.14	0
CWE 659 4.7	0
CWE 660 4.14	125
CWE 660 4.7	68
Code conventions for the Java Programming Language(Oracle)	0
JavaScript 시큐어코딩 가이드 2022	0
MISRA-C 2004	0
MISRA-C 2012	0
MISRA-C 2012 Amendment 2	0
MISRA-C 2012 Amendment 3	0
MISRA-C++ 2008	0
OWASP 2017	7
OWASP 2021	29
Python 시큐어코딩 가이드 2022	0
Rust ANSSI guide v1.0	0
무기체계 소프트웨어 보안약점 점검 목록	227
방위사업청 코딩규칙	0
소프트웨어 보안약점 진단가이드 2021	236
주요정보통신기반시설 취약점 분석·평가 기준	0

.NET framework design guideline

Reference Chapter	Issues
System.Xml 사용법	0
구조체 디자인	0
네임스페이스의 이름	0
리소스 이름 지정	0
매개변수 이름 지정	0
멤버 오버로드	0
보호된 멤버	0

봉인	0
예외 throw	0
예외 및 성능	0
인터페이스 디자인	0
일반 명명 규칙	0
컬렉션	0
클래스와 구조체 간의 선택	0
표준 예외 형식 사용	0

● CWE 658 4.14

Reference Chapter	Issues
119 - Improper Restriction of Operations within the Bounds of a Memory Buffer	0
120 - Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')	0
121 - Stack-based Buffer Overflow	0
122 - Heap-based Buffer Overflow	0
123 - Write-what-where Condition	0
124 - Buffer Underwrite ('Buffer Underflow')	0
125 - Out-of-bounds Read	0
126 - Buffer Over-read	0
127 - Buffer Under-read	0
128 - Wrap-around Error	0
129 - Improper Validation of Array Index	0
131 - Incorrect Calculation of Buffer Size	0
1325 - Improperly Controlled Sequential Memory Allocation	0
1335 - Incorrect Bitwise Shift of Integer	0
134 - Use of Externally-Controlled Format String	0
1341 - Multiple Releases of Same Resource or Handle	0
135 - Incorrect Calculation of Multi-Byte String Length	0
14 - Compiler Removal of Code to Clear Buffers	0
170 - Improper Null Termination	0
188 - Reliance on Data/Memory Layout	0
191 - Integer Underflow (Wrap or Wraparound)	0
192 - Integer Coercion Error	0
194 - Unexpected Sign Extension	0
195 - Signed to Unsigned Conversion Error	0

196 - Unsigned to Signed Conversion Error	0
197 - Numeric Truncation Error	0
242 - Use of Inherently Dangerous Function	0
243 - Creation of chroot Jail Without Changing Working Directory	0
244 - Improper Clearing of Heap Memory Before Release ('Heap Inspection')	0
362 - Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	0
364 - Signal Handler Race Condition	0
366 - Race Condition within a Thread	0
375 - Returning a Mutable Object to an Untrusted Caller	0
401 - Missing Release of Memory after Effective Lifetime	0
415 - Double Free	0
416 - Use After Free	0
457 - Use of Uninitialized Variable	0
462 - Duplicate Key in Associative List (Alist)	0
463 - Deletion of Data Structure Sentinel	0
464 - Addition of Data Structure Sentinel	0
467 - Use of sizeof() on a Pointer Type	0
468 - Incorrect Pointer Scaling	0
469 - Use of Pointer Subtraction to Determine Size	0
476 - NULL Pointer Dereference	0
478 - Missing Default Case in Multiple Condition Expression	0
479 - Signal Handler Use of a Non-reentrant Function	0
480 - Use of Incorrect Operator	0
481 - Assigning instead of Comparing	0
482 - Comparing instead of Assigning	0
483 - Incorrect Block Delimitation	0
484 - Omitted Break Statement in Switch	0
558 - Use of getlogin() in Multithreaded Application	0
560 - Use of umask() with chmod-style Argument	0
562 - Return of Stack Variable Address	0
587 - Assignment of a Fixed Address to a Pointer	0
676 - Use of Potentially Dangerous Function	0
685 - Function Call With Incorrect Number of Arguments	0
690 - Unchecked Return Value to NULL Pointer Dereference	0
704 - Incorrect Type Conversion or Cast	0

733 - Compiler Optimization Removal or Modification of Security-critical Code	0
762 - Mismatched Memory Management Routines	0
783 - Operator Precedence Logic Error	0
785 - Use of Path Manipulation Function without Maximum-sized Buffer	0
787 - Out-of-bounds Write	0
789 - Memory Allocation with Excessive Size Value	0
805 - Buffer Access with Incorrect Length Value	0
806 - Buffer Access Using Size of Source Buffer	0
839 - Numeric Range Comparison Without Minimum Check	0
843 - Access of Resource Using Incompatible Type ('Type Confusion')	0
910 - Use of Expired File Descriptor	0

● CWE 658 4.7

Reference Chapter	Issues
Access of Resource Using Incompatible Type ('Type Confusion') - (843)	0
Addition of Data Structure Sentinel - (464)	0
Assigning instead of Comparing - (481)	0
Assignment of a Fixed Address to a Pointer - (587)	0
Buffer Access Using Size of Source Buffer - (806)	0
Buffer Access with Incorrect Length Value - (805)	0
Buffer Copy without Checking Size of Input ('Classic Buffer Overflow') - (120)	0
Buffer Over-read - (126)	0
Buffer Under-read - (127)	0
Buffer Underwrite ('Buffer Underflow') - (124)	0
Comparing instead of Assigning - (482)	0
Compiler Optimization Removal or Modification of Security-critical Code - (733)	0
Compiler Removal of Code to Clear Buffers - (14)	0
Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition') - (362)	0
Creation of chroot Jail Without Changing Working Directory - (243)	0
Deletion of Data Structure Sentinel - (463)	0
Double Free - (415)	0
Duplicate Key in Associative List (Alist) - (462)	0
Function Call With Incorrect Number of Arguments - (685)	0
Function Call With Incorrect Variable or Reference as Argument - (688)	0

Heap-based Buffer Overflow - (122)	0
Improper Cleanup on Thrown Exception - (460)	0
Improper Clearing of Heap Memory Before Release ('Heap Inspection') - (244)	0
Improper Handling of Length Parameter Inconsistency - (130)	0
Improper Null Termination - (170)	0
Improper Restriction of Operations within the Bounds of a Memory Buffer - (119)	0
Improper Update of Reference Count - (911)	0
Improper Validation of Array Index - (129)	0
Incorrect Block Delimitation - (483)	0
Incorrect Calculation of Buffer Size - (131)	0
Incorrect Calculation of Multi-Byte String Length - (135)	0
Incorrect Pointer Scaling - (468)	0
Incorrect Type Conversion or Cast - (704)	0
Integer Coercion Error - (192)	0
Integer Underflow (Wrap or Wraparound) - (191)	0
Mismatched Memory Management Routines - (762)	0
Missing Default Case in Switch Statement - (478)	0
NULL Pointer Dereference - (476)	0
Numeric Range Comparison Without Minimum Check - (839)	0
Numeric Truncation Error - (197)	0
Omitted Break Statement in Switch - (484)	0
Operator Precedence Logic Error - (783)	0
Out-of-bounds Read - (125)	0
Out-of-bounds Write - (787)	0
Race Condition within a Thread - (366)	0
Reliance on Data/Memory Layout - (188)	0
Return of Pointer Value Outside of Expected Range - (466)	0
Return of Stack Variable Address - (562)	0
Signal Handler Race Condition - (364)	0
Signal Handler Use of a Non-reentrant Function - (479)	0
Signed to Unsigned Conversion Error - (195)	0
Stack-based Buffer Overflow - (121)	0
Unexpected Sign Extension - (194)	0
Unsigned to Signed Conversion Error - (196)	0
Use After Free - (416)	0
Use of Expired File Descriptor - (910)	0

Use of Externally-Controlled Format String - (134)	0
Use of Incorrect Operator - (480)	0
Use of Inherently Dangerous Function - (242)	0
Use of Pointer Subtraction to Determine Size - (469)	0
Use of Potentially Dangerous Function - (676)	0
Use of Uninitialized Variable - (457)	0
Use of getlogin() in Multithreaded Application - (558)	0
Use of sizeof() on a Pointer Type - (467)	0
Use of umask() with chmod-style Argument - (560)	0
Wrap-around Error - (128)	0
Write-what-where Condition - (123)	0

CWE 659 4.14

Reference Chapter	Issues
119 - Improper Restriction of Operations within the Bounds of a Memory Buffer	0
120 - Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')	0
121 - Stack-based Buffer Overflow	0
122 - Heap-based Buffer Overflow	0
123 - Write-what-where Condition	0
124 - Buffer Underwrite ('Buffer Underflow')	0
125 - Out-of-bounds Read	0
126 - Buffer Over-read	0
127 - Buffer Under-read	0
128 - Wrap-around Error	0
129 - Improper Validation of Array Index	0
130 - Improper Handling of Length Parameter Inconsistency	0
131 - Incorrect Calculation of Buffer Size	0
1325 - Improperly Controlled Sequential Memory Allocation	0
1335 - Incorrect Bitwise Shift of Integer	0
134 - Use of Externally-Controlled Format String	0
1341 - Multiple Releases of Same Resource or Handle	0
135 - Incorrect Calculation of Multi-Byte String Length	0
14 - Compiler Removal of Code to Clear Buffers	0
170 - Improper Null Termination	0
188 - Reliance on Data/Memory Layout	0

191 - Integer Underflow (Wrap or Wraparound)	0
192 - Integer Coercion Error	0
194 - Unexpected Sign Extension	0
195 - Signed to Unsigned Conversion Error	0
196 - Unsigned to Signed Conversion Error	0
197 - Numeric Truncation Error	0
242 - Use of Inherently Dangerous Function	0
243 - Creation of chroot Jail Without Changing Working Directory	0
244 - Improper Clearing of Heap Memory Before Release ('Heap Inspection')	0
248 - Uncaught Exception	0
362 - Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	0
364 - Signal Handler Race Condition	0
366 - Race Condition within a Thread	0
374 - Passing Mutable Objects to an Untrusted Method	0
375 - Returning a Mutable Object to an Untrusted Caller	0
396 - Declaration of Catch for Generic Exception	0
397 - Declaration of Throws for Generic Exception	0
401 - Missing Release of Memory after Effective Lifetime	0
415 - Double Free	0
416 - Use After Free	0
457 - Use of Uninitialized Variable	0
462 - Duplicate Key in Associative List (Alist)	0
463 - Deletion of Data Structure Sentinel	0
464 - Addition of Data Structure Sentinel	0
467 - Use of sizeof() on a Pointer Type	0
468 - Incorrect Pointer Scaling	0
469 - Use of Pointer Subtraction to Determine Size	0
476 - NULL Pointer Dereference	0
478 - Missing Default Case in Multiple Condition Expression	0
479 - Signal Handler Use of a Non-reentrant Function	0
480 - Use of Incorrect Operator	0
481 - Assigning instead of Comparing	0
482 - Comparing instead of Assigning	0
483 - Incorrect Block Delimitation	0
484 - Omitted Break Statement in Switch	0

493 - Critical Public Variable Without Final Modifier	0
495 - Private Data Structure Returned From A Public Method	0
496 - Public Data Assigned to Private Array-Typed Field	0
498 - Cloneable Class Containing Sensitive Information	0
500 - Public Static Field Not Marked Final	0
543 - Use of Singleton Pattern Without Synchronization in a Multithreaded Context	0
558 - Use of getlogin() in Multithreaded Application	0
562 - Return of Stack Variable Address	0
587 - Assignment of a Fixed Address to a Pointer	0
676 - Use of Potentially Dangerous Function	0
690 - Unchecked Return Value to NULL Pointer Dereference	0
704 - Incorrect Type Conversion or Cast	0
733 - Compiler Optimization Removal or Modification of Security-critical Code	0
762 - Mismatched Memory Management Routines	0
766 - Critical Data Element Declared Public	0
767 - Access to Critical Private Variable via Public Method	0
783 - Operator Precedence Logic Error	0
785 - Use of Path Manipulation Function without Maximum-sized Buffer	0
787 - Out-of-bounds Write	0
789 - Memory Allocation with Excessive Size Value	0
805 - Buffer Access with Incorrect Length Value	0
806 - Buffer Access Using Size of Source Buffer	0
839 - Numeric Range Comparison Without Minimum Check	0
843 - Access of Resource Using Incompatible Type ('Type Confusion')	0
910 - Use of Expired File Descriptor	0

● CWE 659 4.7

Reference Chapter	Issues
Access of Resource Using Incompatible Type ('Type Confusion') - (843)	0
Access to Critical Private Variable via Public Method - (767)	0
Addition of Data Structure Sentinel - (464)	0
Assigning instead of Comparing - (481)	0
Assignment of a Fixed Address to a Pointer - (587)	0
Buffer Access Using Size of Source Buffer - (806)	0
Buffer Access with Incorrect Length Value - (805)	0

Buffer Copy without Checking Size of Input ('Classic Buffer Overflow') - (120)	0
Buffer Over-read - (126)	0
Buffer Under-read - (127)	0
Buffer Underwrite ('Buffer Underflow') - (124)	0
Comparing instead of Assigning - (482)	0
Compiler Optimization Removal or Modification of Security-critical Code - (733)	0
Compiler Removal of Code to Clear Buffers - (14)	0
Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition') - (362)	0
Creation of chroot Jail Without Changing Working Directory - (243)	0
Critical Public Variable Without Final Modifier - (493)	0
Declaration of Catch for Generic Exception - (396)	0
Declaration of Throws for Generic Exception - (397)	0
Deletion of Data Structure Sentinel - (463)	0
Double Free - (415)	0
Duplicate Key in Associative List (Alist) - (462)	0
Heap-based Buffer Overflow - (122)	0
Improper Cleanup on Thrown Exception - (460)	0
Improper Clearing of Heap Memory Before Release ('Heap Inspection') - (244)	0
Improper Handling of Length Parameter Inconsistency - (130)	0
Improper Null Termination - (170)	0
Improper Restriction of Operations within the Bounds of a Memory Buffer - (119)	0
Improper Update of Reference Count - (911)	0
Improper Validation of Array Index - (129)	0
Incorrect Block Delimitation - (483)	0
Incorrect Calculation of Buffer Size - (131)	0
Incorrect Calculation of Multi-Byte String Length - (135)	0
Incorrect Pointer Scaling - (468)	0
Incorrect Type Conversion or Cast - (704)	0
Integer Coercion Error - (192)	0
Integer Underflow (Wrap or Wraparound) - (191)	0
Mismatched Memory Management Routines - (762)	0
Missing Default Case in Switch Statement - (478)	0
NULL Pointer Dereference - (476)	0
Numeric Range Comparison Without Minimum Check - (839)	0
Numeric Truncation Error - (197)	0

Omitted Break Statement in Switch - (484)	0
Operator Precedence Logic Error - (783)	0
Out-of-bounds Read - (125)	0
Out-of-bounds Write - (787)	0
Passing Mutable Objects to an Untrusted Method - (374)	0
Public Data Assigned to Private Array-Typed Field - (496)	0
Race Condition within a Thread - (366)	0
Reliance on Data/Memory Layout - (188)	0
Return of Pointer Value Outside of Expected Range - (466)	0
Return of Stack Variable Address - (562)	0
Returning a Mutable Object to an Untrusted Caller - (375)	0
Signal Handler Race Condition - (364)	0
Signal Handler Use of a Non-reentrant Function - (479)	0
Signed to Unsigned Conversion Error - (195)	0
Stack-based Buffer Overflow - (121)	0
Uncaught Exception - (248)	0
Unexpected Sign Extension - (194)	0
Unsigned to Signed Conversion Error - (196)	0
Use After Free - (416)	0
Use of Expired File Descriptor - (910)	0
Use of Externally-Controlled Format String - (134)	0
Use of Incorrect Operator - (480)	0
Use of Inherently Dangerous Function - (242)	0
Use of Pointer Subtraction to Determine Size - (469)	0
Use of Potentially Dangerous Function - (676)	0
Use of Uninitialized Variable - (457)	0
Use of getlogin() in Multithreaded Application - (558)	0
Use of sizeof() on a Pointer Type - (467)	0
Wrap-around Error - (128)	0
Write-what-where Condition - (123)	0

CWE 660 4.14

Reference Chapter	Issues
102 - Struts: Duplicate Validation Forms	0
103 - Struts: Incomplete validate() Method Definition	0

104 - Struts: Form Bean Does Not Extend Validation Class	0
106 - Struts: Plug-in Framework not in Use	0
109 - Struts: Validator Turned Off	0
110 - Struts: Validator Without Form Field	0
111 - Direct Use of Unsafe JNI	0
1235 - Incorrect Use of Autoboxing and Unboxing for Performance Critical Operations	0
1335 - Incorrect Bitwise Shift of Integer	0
1336 - Improper Neutralization of Special Elements Used in a Template Engine	0
1341 - Multiple Releases of Same Resource or Handle	1
191 - Integer Underflow (Wrap or Wraparound)	1
192 - Integer Coercion Error	0
197 - Numeric Truncation Error	0
209 - Generation of Error Message Containing Sensitive Information	53
245 - J2EE Bad Practices: Direct Management of Connections	0
246 - J2EE Bad Practices: Direct Use of Sockets	0
248 - Uncaught Exception	0
362 - Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	1
366 - Race Condition within a Thread	2
374 - Passing Mutable Objects to an Untrusted Method	0
375 - Returning a Mutable Object to an Untrusted Caller	0
382 - J2EE Bad Practices: Use of System.exit()	0
383 - J2EE Bad Practices: Direct Use of Threads	0
396 - Declaration of Catch for Generic Exception	0
397 - Declaration of Throws for Generic Exception	20
460 - Improper Cleanup on Thrown Exception	7
470 - Use of Externally-Controlled Input to Select Classes or Code ('Unsafe Reflection')	0
476 - NULL Pointer Dereference	34
478 - Missing Default Case in Multiple Condition Expression	0
481 - Assigning instead of Comparing	0
484 - Omitted Break Statement in Switch	0
486 - Comparison of Classes by Name	0
487 - Reliance on Package-level Scope	0
491 - Public cloneable() Method Without Final ('Object Hijack')	0

492 - Use of Inner Class Containing Sensitive Data	0
493 - Critical Public Variable Without Final Modifier	3
495 - Private Data Structure Returned From A Public Method	0
496 - Public Data Assigned to Private Array-Typed Field	0
498 - Cloneable Class Containing Sensitive Information	0
500 - Public Static Field Not Marked Final	3
502 - Deserialization of Untrusted Data	0
537 - Java Runtime Error Message Containing Sensitive Information	0
567 - Unsynchronized Access to Shared Data in a Multithreaded Context	0
568 - finalize() Method Without super.finalize()	0
572 - Call to Thread run() instead of start()	0
574 - EJB Bad Practices: Use of Synchronization Primitives	0
575 - EJB Bad Practices: Use of AWT Swing	0
576 - EJB Bad Practices: Use of Java I/O	0
577 - EJB Bad Practices: Use of Sockets	0
578 - EJB Bad Practices: Use of Class Loader	0
579 - J2EE Bad Practices: Non-serializable Object Stored in Session	0
580 - clone() Method Without super.clone()	0
581 - Object Model Violation: Just One of Equals and Hashcode Defined	0
582 - Array Declared Public, Final, and Static	0
583 - finalize() Method Declared Public	0
594 - J2EE Framework: Saving Unserializable Objects to Disk	0
595 - Comparison of Object References Instead of Object Contents	0
607 - Public Static Final Field References Mutable Object	0
608 - Struts: Non-private Field in ActionForm Class	0
609 - Double-Checked Locking	0
7 - J2EE Misconfiguration: Missing Custom Error Page	0
766 - Critical Data Element Declared Public	0
917 - Improper Neutralization of Special Elements used in an Expression Language Statement ('Expression Language Injection')	0
95 - Improper Neutralization of Directives in Dynamically Evaluated Code ('Eval Injection')	0

● CWE 660 4.7

Reference Chapter	Issues

Array Declared Public, Final, and Static - (582)	0
Assigning instead of Comparing - (481)	0
Call to Thread run() instead of start() - (572)	0
Cloneable Class Containing Sensitive Information - (498)	0
Comparison of Classes by Name - (486)	0
Comparison of Object References Instead of Object Contents - (595)	0
Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition') - (362)	1
Critical Public Variable Without Final Modifier - (493)	3
Declaration of Catch for Generic Exception - (396)	0
Declaration of Throws for Generic Exception - (397)	20
Deserialization of Untrusted Data - (502)	0
Direct Use of Unsafe JNI - (111)	0
Double-Checked Locking - (609)	0
EJB Bad Practices: Use of AWT Swing - (575)	0
EJB Bad Practices: Use of Java I/O - (576)	0
EJB Bad Practices: Use of Sockets - (577)	0
Finalize() Method Without super.finalize() - (568)	0
Improper Cleanup on Thrown Exception - (460)	7
Improper Neutralization of Directives in Dynamically Evaluated Code ('Eval Injection') - (95)	0
J2EE Bad Practices: Direct Management of Connections - (245)	0
J2EE Bad Practices: Direct Use of Sockets - (246)	0
J2EE Bad Practices: Direct Use of Threads - (383)	0
J2EE Bad Practices: Use of System.exit() - (382)	0
NULL Pointer Dereference - (476)	34
Numeric Truncation Error - (197)	0
Object Model Violation: Just One of Equals and Hashcode Defined - (581)	0
Omitted Break Statement in Switch - (484)	0
Passing Mutable Objects to an Untrusted Method - (374)	0
Public Data Assigned to Private Array-Typed Field - (496)	0
Public Static Field Not Marked Final - (500)	3
Public Static Final Field References Mutable Object - (607)	0
Reliance on Package-level Scope - (487)	0
Returning a Mutable Object to an Untrusted Caller - (375)	0
Uncaught Exception - (248)	0

Use of Inner Class Containing Sensitive Data - (492)	0

● Code conventions for the Java Programming Language(Oracle)

Reference Chapter	Issues
04.1 Line Length	0
04.2 Wrapping Lines	0
05.1.1 Block Comments	0
05.1.2 Single-Line Comments	0
05.1.3 Trailing Comments	0
05.1.4 End-Of-Line Comments	0
05.2 Documentation Comments	0
06.1 Number Per Line	0
06.2 Initialization	0
06.3 Placement	0
06.4 Class and Interface Declarations	0
07.1 Simple Statements	0
07.2 Compound Statements	0
07.3 return Statements	0
07.4 if, if-else, if else-if else Statements	0
07.5 for Statements	0
07.6 while Statements	0
07.7 do-while Statements	0
07.8 switch Statements	0
07.9 try-catch Statements	0
08.1 Blank Lines	0
08.2 Blank Spaces	0
09.1 Package	0
09.2 Classes or Interface	0
09.3 Methods	0
09.4 Variables	0
09.5 Constants	0
10.1 Providing Access to Instance and Class Variables	0
10.2 Refferring to Class Variables and Methods	0
10.3 Constants	0
10.4 Variable Assignments	0

10.5.1 Parentheses	0
10.5.2 Returning Values	0
10.5.3 Expressions before '?' in the Conditional Operator	0

JavaScript 시큐어코딩 가이드 2022

Reference Chapter	Issues
01.01. SQL 삽입	0
01.02. 코드 삽입	0
01.03. 경로 조작 및 자원 삽입	0
01.04. 크로스사이트 스크립트(XSS)	0
01.05. 운영체제 명령어 삽입	0
01.08. 부적절한 XML 외부 개체 참조	0
01.11. 크로스사이트 요청 위조(CSRF)	0
02.04. 취약한 암호화 알고리즘 사용	0
02.07. 충분하지 않은 키 길이 사용	0
02.08. 적절하지 않은 난수 값 사용	0
02.14. 솔트 없이 일방향 해쉬 함수 사용	0
03.01. 종료되지 않는 반복문 또는 재귀 함수	0
04.01. 오류 메시지 정보 노출	0
06.02. 제거되지 않고 남은 디버그 코드	0

MISRA-C 2004

Reference Chapter	Issues
1.02 (Required) : No reliance shall be placed on undefined or unspecified behaviour.	0
1.04 (Required) : The compiler/linker shall be checked to ensure that 31 character significance and case sensitivity are supported for external identifiers.	0
10.03 (Required) : The value of a complex expression of integer type may only be cast to a type that is narrower and of the same signedness as the underlying type of the expression.	0
10.04 (Required) : The value of a complex expression of floating type may only be cast to a narrower floating type.	0
10.05 (Required) : If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand.	0
10.06 (Required) : A "U" suffix shall be applied to all constants of unsigned type.	0

11.01 (Required) : Conversions shall not be performed between a pointer to a function and any type other than an integral type.	0
11.02 (Required) : Conversions shall not be performed between a pointer to object and any type other than an integral type, another pointer to object type or a pointer to void.	0
11.03 (Advisory) : A cast should not be performed between a pointer type and an integral type.	0
11.04 (Advisory) : A cast should not be performed between a pointer to object type and a different pointer to object type.	0
11.05 (Required) : A cast shall not be performed that removes any const or volatile qualification from the type addressed by a pointer.	0
12.01 (Advisory) : Limited dependence should be placed on C's operator precedence rules in expressions.	0
12.02 (Required) : The value of an expression shall be the same under any order of evaluation that the standard permits.	0
12.03 (Required) : The sizeof operator shall not be used on expressions that contain side effects.	0
12.04 (Required) : The right hand operand of a logical && or || operator shall not contain side effects.	0
12.05 (Required) : The operands of a logical && or || shall be primary-expressions.	0
12.06 (Advisory) : The operands of logical operators (&&, || and !) should be effectively Boolean. Expressions that are effectively Boolean should not be used as operands to operators other than (&&, || and !).	0
12.07 (Required) : Bitwise operators shall not be applied to operands whose underlying type is signed.	0
12.08 (Required) : The right hand operand of a shift operator shall lie between zero and one less than the width in bits of the underlying type of the left hand operand.	0
12.09 (Required) : The unary minus operator shall not be applied to an expression whose underlying type is unsigned.	0
12.10 (Required) : The comma operator shall not be used.	0
12.11 (Advisory) : Evaluation of constant unsigned integer expressions should not lead to wrap-around.	0
12.12 (Required) : The underlying bit representations of floating-point values shall not be used.	0
12.13 (Advisory) : The increment (++) and decrement (--) operators should not be mixed with other operators in an expression.	0
13.01 (Required) : Assignment operators shall not be used in expressions that yield a Boolean value.	0

13.02 (Advisory) : Tests of a value against zero should be made explicit, unless the operand is effectively Boolean.	0
13.03 (Required) : Floating-point expressions shall not be tested for equality or inequality.	0
13.04 (Required) : The controlling expression of a for statement shall not contain any objects of floating type.	0
13.05 (Required) : The three expressions of a for statement shall be concerned only with loop control.	0
13.06 (Required) : Numeric variables being used within a for loop for iteration counting shall not be modified in the body of the loop.	0
13.07 (Required) : Boolean operations whose results are invariant shall not be permitted.	0
14.01 (Required) : There shall be no unreachable code.	0
14.02 (Required) : All non-null statements shall either : a) have at least one side-effect however executed, or b) cause control flow to change.	0
14.03 (Required) : Before preprocessing, a null statement shall only occur on a line by itself; it may be followed by a comment provided that the first character following the null statement is a white-space character.	0
14.04 (Required) : The goto statement shall not be used.	0
14.05 (Required) : The continue statement shall not be used.	0
14.06 (Required) : For any iteration statement there shall be at most one break statement used for loop termination.	0
14.07 (Required) : A function shall have a single point of exit at the end of the function.	0
14.08 (Required) : The statement forming the body of a switch, while, do ... while or for statement shall be a compound statement.	0
14.09 (Required) : An if (expression) construct shall be followed by a compound statement. The else keyword shall be followed by either a compound statement, or another if statement.	0
14.10 (Required) : All if ... else if constructs shall be terminated with an else clause.	0
15.01 (Required) : A switch label shall only be used when the most closely-enclosing compound statement is the body of a switch statement.	0
15.02 (Required) : An unconditional break statement shall terminate every non-empty switch clause.	0
15.03 (Required) : The final clause of a switch statement shall be the default clause.	0
15.04 (Required) : A switch expression shall not represent a value that is effectively Boolean.	0
15.05 (Required) : Every switch statement shall have at least one case clause.	0

16.01 (Required) : Functions shall not be defined with a variable number of arguments.	0
16.02 (Required) : Functions shall not call themselves, either directly or indirectly.	0
16.03 (Required) : Identifiers shall be given for all of the parameters in a function prototype declaration.	0
16.04 (Required) : The identifiers used in the declaration and definition of a function shall be identical.	0
16.05 (Required) : The identifiers used in the declaration and definition of a function shall be identical.	0
16.06 (Required) : The number of arguments passed to a function shall match the number of parameters.	0
16.07 (Advisory) : A pointer parameter in a function prototype should be declared as pointer to const if the pointer is not used to modify the addressed object.	0
16.08 (Required) : All exit paths from a function with non-void return type shall have an explicit return statement with an expression.	0
16.09 (Required) : A function identifier shall only be used with either a preceding &, or with a parenthesised parameter list, which may be empty.	0
16.10 (Required) : If a function returns error information, then that error information shall be tested.	0
17.01 (Required) : Pointer arithmetic shall only be applied to pointers that address an array or array element.	0
17.02 (Required) : Pointer subtraction shall only be applied to pointers that address elements of the same array.	0
17.03 (Required) : >, >=, <, <= shall not be applied to pointer types except where they point to the same array.	0
17.04 (Required) : Array indexing shall be the only allowed form of pointer arithmetic.	0
17.05 (Advisory) : The declaration of objects should contain no more than 2 levels of pointer indirection.	0
17.06 (Required) : The address of an object with automatic storage shall not be assigned to another object that may persist after the first object has ceased to exist.	0
18.01 (Required) : All structure and union types shall be complete at the end of a translation unit.	0
18.02 (Required) : An object shall not be assigned to an overlapping object.	0
18.04 (Required) : Unions shall not be used.	0
19.01 (Advisory) : #include statements in a file should only be preceded by other preprocessor directives or comments.	0
19.02 (Advisory) : Non-standard characters should not occur in header file names in	

#include directives.	0
19.03 (Required) : The #include directive shall be followed by either a <filename> or "filename" sequence.	0
19.04 (Required) : C macros shall only expand to a braced initialiser, a constant, a parenthesised expression, a type qualifier, a storage class specifier, or a do-while-zero construct.	0
19.05 (Required) : Macros shall not be #define'd or #undef'd within a block.	0
19.06 (Required) : #undef shall not be used.	0
19.07 (Advisory) : A function should be used in preference to a function-like macro.	0
19.08 (Required) : A function-like macro shall not be invoked without all of its arguments.	0
19.09 (Required) : Arguments to a function-like macro shall not contain tokens that look like preprocessing directives.	0
19.10 (Required) : In the definition of a function-like macro each instance of a parameter shall be enclosed in parentheses unless it is used as the operand of # or ##.	0
19.11 (Required) : All macro identifiers in preprocessor directives shall be defined before use, except in #ifdef and #ifndef preprocessor directives and the defined() operator.	0
19.12 (Required) : There shall be at most one occurrence of the # or ## operators in a single macro definition.	0
19.13 (Advisory) : The # and ## operators should not be used.	0
19.14 (Required) : The defined preprocessor operator shall only be used in one of the two standard forms.	0
19.15 (Required) : Precautions shall be taken in order to prevent the contents of a header file being included twice.	0
19.16 (Required) : Preprocessing directives shall be syntactically meaningful even when excluded by the preprocessor.	0
2.01 (Required) : Assembly language shall be encapsulated and isolated.	0
2.02 (Required) : Source code shall only use /* ... */ style comments.	0
2.03 (Required) : The character sequence /* shall not be used within a comment.	0
20.01 (Required) : Reserved identifiers, macros and functions in the standard library, shall not be defined, redefined or undefined.	0
20.02 (Required) : The names of standard library macros, objects and functions shall not be reused.	0
20.04 (Required) : Dynamic heap memory allocation shall not be used.	0
20.05 (Required) : The error indicator errno shall not be used.	0

20.06 (Required) : The macro offsetof, in library <stddef.h>, shall not be used.	0
20.07 (Required) : The setjmp macro and the longjmp function shall not be used.	0
20.08 (Required) : The signal handling facilities of <signal.h> shall not be used.	0
20.09 (Required) : The input/output library <stdio.h> shall not be used in production code.	0
20.10 (Required) : The library functions atof, atoi and atol from library <stdlib.h> shall not be used.	0
20.11 (Required) : The library functions abort, exit, getenv and system from library <stdlib.h> shall not be used.	0
20.12 (Required) : The time handling functions of library <time.h> shall not be used.	0
21.1 (Required) : Minimisation of run-time failures shall be ensured by the use of at least one of a) static analysis tools/techniques; b) dynamic analysis tools/techniques; c) explicit coding of checks to handle run-time faults.	0
3.04 (Required) : All uses of the #pragma directive shall be documented and explained.	0
3.05 (Required) : If it is being relied upon, the implementation defined behaviour and packing of bitfields shall be documented.	0
4.01 (Required) : Only those escape sequences that are defined in the ISO C standard shall be used.	0
4.02 (Required) : Trigraphs shall not be used.	0
5.01 (Required) : Identifiers (internal and external) shall not rely on the significance of more than 31 characters.	0
5.02 (Required) : Identifiers in an inner scope shall not use the same name as an identifier in an outer scope, and therefore hide that identifier.	0
5.03 (Required) : A typedef name shall be a unique identifier.	0
5.04 (Required) : A tag name shall be a unique identifier.	0
5.05 (Advisory) : No object or function identifier with static storage duration should be reused.	0
5.06 (Advisory) : No identifier in one name space should have the same spelling as an identifier in another name space, with the exception of structure and union member names.	0
5.07 (Advisory) : No identifier name should be reused.	0
6.01 (Required) : The plain char type shall be used only for the storage and use of character values.	0
6.02 (Required) : Signed and unsigned char type shall be used only for the storage and use of numeric values.	0
6.03 (Advisory) : Typedefs that indicate size and signedness should be used in place of the basic types.	0

6.04 (Required) : Bit fields shall only be defined to be of type unsigned int or signed int.	0
6.05 (Required) : Bit fields of type signed int shall be at least 2 bits long.	0
7.01 (Required) : Octal constants (other than zero) and octal escape sequences shall not be used.	0
8.02 (Required) : Whenever an object or function is declared or defined, its type shall be explicitly stated.	0
8.03 (Required) : For each function parameter the type given in the declaration and definition shall be identical, and the return types shall also be identical.	0
8.04 (Required) : If objects or functions are declared more than once their types shall be compatible.	0
8.05 (Required) : There shall be no definitions of objects or functions in a header file.	0
8.06 (Required) : Functions shall be declared at file scope.	0
8.07 (Required) : Objects shall be defined at block scope if they are only accessed from within a single function.	0
8.08 (Required) : An external object or function shall be declared in one and only one file.	0
8.09 (Required) : An identifier with external linkage shall have exactly one external definition.	0
8.10 (Required) : All declarations and definitions of objects or functions at file scope shall have internal linkage unless external linkage is required.	0
8.11 (Required) : The static storage class specifier shall be used in definitions and declarations of objects and functions that have internal linkage.	0
8.12 (Required) : When an array is declared with external linkage, its size shall be stated explicitly or defined implicitly by initialisation.	0
9.01 (Required) : All automatic variables shall have been assigned a value before being used.	0
9.02 (Required) : Braces shall be used to indicate and match the structure in the non-zero initialisation of arrays and structures.	0
9.03 (Required) : In an enumerator list, the "=" construct shall not be used to explicitly initialise members other than the first, unless all items are explicitly initialised.	0

● MISRA-C 2012

Reference Chapter	Issues
Directives 1.1	0
Directives 4.1	0

Directives 4.10	0
Directives 4.12	0
Directives 4.14	0
Directives 4.3	0
Directives 4.4	0
Directives 4.5	0
Directives 4.6	0
Directives 4.7	0
Directives 4.8	0
Directives 4.9	0
Rule 1.1	0
Rule 1.2	0
Rule 1.3	0
Rule 10.1	0
Rule 10.2	0
Rule 10.3	0
Rule 10.4	0
Rule 10.5	0
Rule 10.6	0
Rule 10.7	0
Rule 10.8	0
Rule 11.1	0
Rule 11.2	0
Rule 11.3	0
Rule 11.4	0
Rule 11.5	0
Rule 11.6	0
Rule 11.7	0
Rule 11.8	0
Rule 11.9	0
Rule 12.1	0
Rule 12.2	0
Rule 12.3	0
Rule 12.4	0
Rule 12.5	0
Rule 13.1	0

Rule 13.2	0
Rule 13.3	0
Rule 13.4	0
Rule 13.5	0
Rule 13.6	0
Rule 14.1	0
Rule 14.2	0
Rule 14.3	0
Rule 14.4	0
Rule 15.1	0
Rule 15.2	0
Rule 15.3	0
Rule 15.4	0
Rule 15.5	0
Rule 15.6	0
Rule 15.7	0
Rule 16.1	0
Rule 16.2	0
Rule 16.3	0
Rule 16.4	0
Rule 16.5	0
Rule 16.6	0
Rule 16.7	0
Rule 17.1	0
Rule 17.2	0
Rule 17.3	0
Rule 17.4	0
Rule 17.5	0
Rule 17.6	0
Rule 17.7	0
Rule 17.8	0
Rule 18.1	0
Rule 18.2	0
Rule 18.3	0
Rule 18.4	0
Rule 18.5	0

Rule 18.6	0
Rule 18.7	0
Rule 18.8	0
Rule 19.1	0
Rule 19.2	0
Rule 2.1	0
Rule 2.2	0
Rule 2.3	0
Rule 2.4	0
Rule 2.5	0
Rule 2.6	0
Rule 2.7	0
Rule 20.01	0
Rule 20.02	0
Rule 20.03	0
Rule 20.04	0
Rule 20.05	0
Rule 20.06	0
Rule 20.07	0
Rule 20.08	0
Rule 20.09	0
Rule 20.10	0
Rule 20.11	0
Rule 20.12	0
Rule 20.13	0
Rule 21.01	0
Rule 21.02	0
Rule 21.03	0
Rule 21.04	0
Rule 21.05	0
Rule 21.06	0
Rule 21.07	0
Rule 21.08	0
Rule 21.09	0
Rule 21.10	0
Rule 21.11	0

Rule 21.12	0
Rule 21.16	0
Rule 21.17	0
Rule 21.18	0
Rule 21.21	0
Rule 22.01	0
Rule 22.02	0
Rule 22.03	0
Rule 22.04	0
Rule 22.05	0
Rule 22.06	0
Rule 22.08	0
Rule 3.1	0
Rule 3.2	0
Rule 4.1	0
Rule 4.2	0
Rule 5.1	0
Rule 5.2	0
Rule 5.3	0
Rule 5.4	0
Rule 5.5	0
Rule 5.6	0
Rule 5.7	0
Rule 5.8	0
Rule 5.9	0
Rule 6.1	0
Rule 6.2	0
Rule 7.1	0
Rule 7.2	0
Rule 7.3	0
Rule 7.4	0
Rule 8.01	0
Rule 8.02	0
Rule 8.03	0
Rule 8.04	0
Rule 8.05	0

Rule 8.06	0
Rule 8.07	0
Rule 8.08	0
Rule 8.09	0
Rule 8.10	0
Rule 8.11	0
Rule 8.12	0
Rule 8.13	0
Rule 8.14	0
Rule 9.1	0
Rule 9.2	0
Rule 9.3	0
Rule 9.4	0
Rule 9.5	0

MISRA-C 2012 Amendment 2

Reference Chapter	Issues
Directives 1.1	0
Directives 4.1	0
Directives 4.3	0
Directives 4.4	0
Directives 4.5	0
Directives 4.6	0
Directives 4.7	0
Directives 4.8	0
Directives 4.9	0
Rule 1.1	0
Rule 1.2	0
Rule 1.3	0
Rule 1.4	0
Rule 10.1	0
Rule 10.2	0
Rule 10.3	0
Rule 10.4	0
Rule 10.5	0

Rule 10.6	0
Rule 10.7	0
Rule 10.8	0
Rule 11.1	0
Rule 11.2	0
Rule 11.3	0
Rule 11.4	0
Rule 11.5	0
Rule 11.6	0
Rule 11.7	0
Rule 11.8	0
Rule 11.9	0
Rule 12.1	0
Rule 12.2	0
Rule 12.3	0
Rule 12.4	0
Rule 12.5	0
Rule 13.1	0
Rule 13.2	0
Rule 13.3	0
Rule 13.4	0
Rule 13.5	0
Rule 13.6	0
Rule 14.1	0
Rule 14.2	0
Rule 14.3	0
Rule 14.4	0
Rule 15.1	0
Rule 15.2	0
Rule 15.3	0
Rule 15.4	0
Rule 15.5	0
Rule 15.6	0
Rule 15.7	0
Rule 16.1	0
Rule 16.2	0

Rule 16.3	0
Rule 16.4	0
Rule 16.5	0
Rule 16.6	0
Rule 16.7	0
Rule 17.1	0
Rule 17.2	0
Rule 17.3	0
Rule 17.4	0
Rule 17.5	0
Rule 17.6	0
Rule 17.7	0
Rule 17.8	0
Rule 18.1	0
Rule 18.2	0
Rule 18.3	0
Rule 18.4	0
Rule 18.5	0
Rule 18.6	0
Rule 18.7	0
Rule 18.8	0
Rule 19.1	0
Rule 19.2	0
Rule 2.1	0
Rule 2.2	0
Rule 2.3	0
Rule 2.4	0
Rule 2.5	0
Rule 2.6	0
Rule 2.7	0
Rule 20.01	0
Rule 20.02	0
Rule 20.03	0
Rule 20.04	0
Rule 20.05	0
Rule 20.06	0

Rule 20.07	0
Rule 20.08	0
Rule 20.09	0
Rule 20.10	0
Rule 20.11	0
Rule 20.12	0
Rule 20.13	0
Rule 21.01	0
Rule 21.02	0
Rule 21.03	0
Rule 21.04	0
Rule 21.05	0
Rule 21.06	0
Rule 21.07	0
Rule 21.08	0
Rule 21.09	0
Rule 21.10	0
Rule 21.11	0
Rule 21.12	0
Rule 21.13	0
Rule 21.14	0
Rule 21.15	0
Rule 21.16	0
Rule 21.17	0
Rule 21.18	0
Rule 21.19	0
Rule 21.20	0
Rule 21.21	0
Rule 22.01	0
Rule 22.02	0
Rule 22.03	0
Rule 22.04	0
Rule 22.05	0
Rule 22.06	0
Rule 22.07	0
Rule 22.08	0

Rule 22.09	0
Rule 22.10	0
Rule 3.1	0
Rule 3.2	0
Rule 4.1	0
Rule 4.2	0
Rule 5.1	0
Rule 5.2	0
Rule 5.3	0
Rule 5.4	0
Rule 5.5	0
Rule 5.6	0
Rule 5.7	0
Rule 5.8	0
Rule 5.9	0
Rule 6.1	0
Rule 6.2	0
Rule 7.1	0
Rule 7.2	0
Rule 7.3	0
Rule 7.4	0
Rule 8.01	0
Rule 8.02	0
Rule 8.03	0
Rule 8.04	0
Rule 8.05	0
Rule 8.06	0
Rule 8.07	0
Rule 8.08	0
Rule 8.09	0
Rule 8.10	0
Rule 8.11	0
Rule 8.12	0
Rule 8.13	0
Rule 8.14	0
Rule 9.1	0

Rule 9.2	0
Rule 9.3	0
Rule 9.4	0
Rule 9.5	0

MISRA-C 2012 Amendment 3

Reference Chapter	Issues
Directives 1.1	0
Directives 4.1	0
Directives 4.10	0
Directives 4.12	0
Directives 4.14	0
Directives 4.3	0
Directives 4.4	0
Directives 4.5	0
Directives 4.6	0
Directives 4.7	0
Directives 4.8	0
Directives 4.9	0
Rule 1.1	0
Rule 1.2	0
Rule 1.3	0
Rule 1.4	0
Rule 10.1	0
Rule 10.2	0
Rule 10.3	0
Rule 10.4	0
Rule 10.5	0
Rule 10.6	0
Rule 10.7	0
Rule 10.8	0
Rule 11.1	0
Rule 11.2	0
Rule 11.3	0
Rule 11.4	0

Rule 11.5	0
Rule 11.6	0
Rule 11.7	0
Rule 11.8	0
Rule 11.9	0
Rule 12.1	0
Rule 12.2	0
Rule 12.3	0
Rule 12.4	0
Rule 12.5	0
Rule 13.1	0
Rule 13.2	0
Rule 13.3	0
Rule 13.4	0
Rule 13.5	0
Rule 13.6	0
Rule 14.1	0
Rule 14.2	0
Rule 14.3	0
Rule 14.4	0
Rule 15.1	0
Rule 15.2	0
Rule 15.3	0
Rule 15.4	0
Rule 15.5	0
Rule 15.6	0
Rule 15.7	0
Rule 16.1	0
Rule 16.2	0
Rule 16.3	0
Rule 16.4	0
Rule 16.5	0
Rule 16.6	0
Rule 16.7	0
Rule 17.1	0
Rule 17.2	0

Rule 17.3	0
Rule 17.4	0
Rule 17.5	0
Rule 17.6	0
Rule 17.7	0
Rule 17.8	0
Rule 18.1	0
Rule 18.2	0
Rule 18.3	0
Rule 18.4	0
Rule 18.5	0
Rule 18.6	0
Rule 18.7	0
Rule 18.8	0
Rule 19.1	0
Rule 19.2	0
Rule 2.1	0
Rule 2.2	0
Rule 2.3	0
Rule 2.4	0
Rule 2.5	0
Rule 2.6	0
Rule 2.7	0
Rule 20.01	0
Rule 20.02	0
Rule 20.03	0
Rule 20.04	0
Rule 20.05	0
Rule 20.06	0
Rule 20.07	0
Rule 20.08	0
Rule 20.09	0
Rule 20.10	0
Rule 20.11	0
Rule 20.12	0
Rule 20.13	0

Rule 21.01	0
Rule 21.02	0
Rule 21.03	0
Rule 21.04	0
Rule 21.05	0
Rule 21.06	0
Rule 21.07	0
Rule 21.08	0
Rule 21.09	0
Rule 21.10	0
Rule 21.11	0
Rule 21.12	0
Rule 21.13	0
Rule 21.14	0
Rule 21.15	0
Rule 21.16	0
Rule 21.17	0
Rule 21.18	0
Rule 21.19	0
Rule 21.20	0
Rule 21.21	0
Rule 22.01	0
Rule 22.02	0
Rule 22.03	0
Rule 22.04	0
Rule 22.05	0
Rule 22.06	0
Rule 22.07	0
Rule 22.08	0
Rule 22.09	0
Rule 22.10	0
Rule 3.1	0
Rule 3.2	0
Rule 4.1	0
Rule 4.2	0
Rule 5.1	0

Rule 5.2	0
Rule 5.3	0
Rule 5.4	0
Rule 5.5	0
Rule 5.6	0
Rule 5.7	0
Rule 5.8	0
Rule 5.9	0
Rule 6.1	0
Rule 6.2	0
Rule 7.1	0
Rule 7.2	0
Rule 7.3	0
Rule 7.4	0
Rule 8.01	0
Rule 8.02	0
Rule 8.03	0
Rule 8.04	0
Rule 8.05	0
Rule 8.06	0
Rule 8.07	0
Rule 8.08	0
Rule 8.09	0
Rule 8.10	0
Rule 8.11	0
Rule 8.12	0
Rule 8.13	0
Rule 8.14	0
Rule 9.1	0
Rule 9.2	0
Rule 9.3	0
Rule 9.4	0
Rule 9.5	0

● MISRA-C++ 2008

Reference Chapter	Issues
Rule 0-1-1	0
Rule 8-3-1	0
Rule0-1-10	0
Rule0-1-11	0
Rule0-1-12	0
Rule0-1-3	0
Rule0-1-4	0
Rule0-1-5	0
Rule0-1-6	0
Rule0-1-7	0
Rule0-1-8	0
Rule0-1-9	0
Rule0-2-1	0
Rule0-3-1	0
Rule10-1-1	0
Rule10-1-2	0
Rule10-1-3	0
Rule10-3-1	0
Rule10-3-2	0
Rule10-3-3	0
Rule11-0-1	0
Rule12-1-1	0
Rule12-1-2	0
Rule12-1-3	0
Rule12-8-1	0
Rule12-8-2	0
Rule14-5-1	0
Rule14-5-2	0
Rule14-5-3	0
Rule14-6-1	0
Rule14-6-2	0
Rule14-7-1	0
Rule14-7-3	0
Rule14-8-1	0
Rule14-8-2	0

Rule15-0-1	0
Rule15-0-2	0
Rule15-1-1	0
Rule15-1-2	0
Rule15-1-3	0
Rule15-3-1	0
Rule15-3-2	0
Rule15-3-3	0
Rule15-3-4	0
Rule15-3-5	0
Rule15-3-6	0
Rule15-3-7	0
Rule15-4-1	0
Rule15-5-1	0
Rule15-5-2	0
Rule15-5-3	0
Rule16-0-1	0
Rule16-0-2	0
Rule16-0-3	0
Rule16-0-4	0
Rule16-0-5	0
Rule16-0-6	0
Rule16-0-7	0
Rule16-0-8	0
Rule16-1-1	0
Rule16-2-1	0
Rule16-2-2	0
Rule16-2-3	0
Rule16-2-4	0
Rule16-2-5	0
Rule16-2-6	0
Rule16-3-1	0
Rule16-3-2	0
Rule17-0-1	0
Rule17-0-2	0
Rule17-0-3	0

Rule17-0-5	0
Rule18-0-1	0
Rule18-0-2	0
Rule18-0-3	0
Rule18-0-4	0
Rule18-0-5	0
Rule18-2-1	0
Rule18-4-1	0
Rule18-7-1	0
Rule19-3-1	0
Rule2-10-1	0
Rule2-10-2	0
Rule2-10-3	0
Rule2-10-4	0
Rule2-10-5	0
Rule2-10-6	0
Rule2-13-1	0
Rule2-13-2	0
Rule2-13-3	0
Rule2-13-4	0
Rule2-13-5	0
Rule2-3-1	0
Rule2-5-1	0
Rule2-7-1	0
Rule2-7-2	0
Rule2-7-3	0
Rule27-0-1	0
Rule3-1-1	0
Rule3-1-2	0
Rule3-1-3	0
Rule3-2-1	0
Rule3-2-2	0
Rule3-2-3	0
Rule3-2-4	0
Rule3-3-1	0
Rule3-3-2	0

Rule3-4-1	0
Rule3-9-1	0
Rule3-9-2	0
Rule3-9-3	0
Rule4-10-1	0
Rule4-10-2	0
Rule4-5-1	0
Rule4-5-2	0
Rule4-5-3	0
Rule5-0-1	0
Rule5-0-10	0
Rule5-0-11	0
Rule5-0-12	0
Rule5-0-13	0
Rule5-0-14	0
Rule5-0-15	0
Rule5-0-16	0
Rule5-0-17	0
Rule5-0-18	0
Rule5-0-19	0
Rule5-0-2	0
Rule5-0-20	0
Rule5-0-21	0
Rule5-0-3	0
Rule5-0-4	0
Rule5-0-5	0
Rule5-0-6	0
Rule5-0-7	0
Rule5-0-8	0
Rule5-0-9	0
Rule5-14-1	0
Rule5-18-1	0
Rule5-19-1	0
Rule5-2-1	0
Rule5-2-10	0
Rule5-2-11	0

Rule5-2-12	0
Rule5-2-2	0
Rule5-2-3	0
Rule5-2-4	0
Rule5-2-5	0
Rule5-2-6	0
Rule5-2-7	0
Rule5-2-8	0
Rule5-2-9	0
Rule5-3-1	0
Rule5-3-2	0
Rule5-3-3	0
Rule5-3-4	0
Rule5-8-1	0
Rule6-2-1	0
Rule6-2-2	0
Rule6-2-3	0
Rule6-3-1	0
Rule6-4-1	0
Rule6-4-2	0
Rule6-4-3	0
Rule6-4-4	0
Rule6-4-5	0
Rule6-4-6	0
Rule6-4-7	0
Rule6-4-8	0
Rule6-5-1	0
Rule6-5-2	0
Rule6-5-3	0
Rule6-5-4	0
Rule6-5-5	0
Rule6-5-6	0
Rule6-6-1	0
Rule6-6-2	0
Rule6-6-3	0
Rule6-6-4	0

Rule6-6-5	0
Rule7-1-1	0
Rule7-1-2	0
Rule7-2-1	0
Rule7-3-1	0
Rule7-3-2	0
Rule7-3-3	0
Rule7-3-4	0
Rule7-3-5	0
Rule7-3-6	0
Rule7-4-2	0
Rule7-4-3	0
Rule7-5-1	0
Rule7-5-2	0
Rule7-5-3	0
Rule7-5-4	0
Rule8-0-1	0
Rule8-4-1	0
Rule8-4-2	0
Rule8-4-3	0
Rule8-4-4	0
Rule8-5-1	0
Rule8-5-2	0
Rule8-5-3	0
Rule9-3-1	0
Rule9-3-2	0
Rule9-3-3	0
Rule9-5-1	0
Rule9-6-1	0
Rule9-6-2	0
Rule9-6-3	0
Rule9-6-4	0

OWASP 2017

Reference Chapter	Issues

A1-Injection	7
A2-Broken Authentication	0
A3-Sensitive Data Exposure	0
A5-Broken Access Control	0
A6-Security Misconfiguration	0

OWASP 2021

Reference Chapter	Issues
A03 Injection	29
A05 Security Misconfiguration	0
A07 Identification and Authentication Failures	0

Python 시큐어코딩 가이드 2022

Reference Chapter	Issues
01.01. SQL 삽입	0
01.02. 코드 삽입	0
01.03. 경로 조작 및 자원 삽입	0
01.04. 크로스사이트 스크립트(XSS)	0
01.05. 운영체제 명령어 삽입	0
01.06. 위험한 형식 파일 업로드	0
01.07. 신뢰되지 않은 URL주소로 자동접속 연결	0
01.08. 부적절한 XML 외부 개체 참조	0
01.09. XML 삽입	0
01.10. LDAP 삽입	0
01.11. 크로스사이트 요청 위조(CSRF)	0
01.12. 서버사이드 요청 위조	0
01.13. HTTP 응답분할	0
01.14. 보안기능 결정에 사용되는 부적절한 입력값	0
01.15. 포맷 스트링 삽입	0
02.01. 적절한 인증 없는 중요 기능 허용	0
02.03. 중요한 자원에 대한 잘못된 권한 설정	0
02.04. 취약한 암호화 알고리즘 사용	0
02.06. 하드코드된 중요정보	0
02.07. 충분하지 않은 키 길이 사용	0

02.08. 적절하지 않은 난수 값 사용	0
02.09. 취약한 비밀번호 허용	0
02.10. 사용자 하드디스크에 저장되는 쿠키를 통한 정보 노출	0
02.11. 주석문 안에 포함된 시스템 주요정보	0
02.12. 솔트 없이 일방향 해쉬 함수 사용	0
02.13. 무결성 검사없는 코드 다운로드	0
03.01. 경쟁조건: 검사시점과 사용시점(TOCTOU)	0
03.02. 종료되지 않는 반복문 또는 재귀 함수	0
04.01. 오류 메시지 정보노출	0
04.02. 오류상황 대응 부재	0
04.03. 부적절한 예외 처리	0
05.01. Null Pointer 역참조	0
05.02. 부적절한 자원 해제	0
05.03. 신뢰할 수 없는 데이터의 역직렬화	0
06.02. 제거되지 않고 남은 디버그 코드	0
06.03. Public 메소드로부터 반환된 Private 배열	0
06.04. Private 배열에 Public 데이터 할당	0

Rust ANSSI guide v1.0

Reference Chapter	Issues
R10 RULE - Don't use unsafe blocks	0
R11 RULE - Use appropriate arithmetic operations regarding potential overflows	0
R13 RECO - Use the ? operator and do not use the try! macro	0
R14 RULE - Don't use functions that can cause panic!	0
R15 RULE - Test properly array indexing or use the get method	0
R16 RULE - Handle correctly panic! in FFI	0
R17 RULE - Do not use forget	0
R19 RULE - Do not leak memory	0
R2 RULE - Keep default values for critical variables in cargo profiles	0
R20 RULE - Do release value wrapped in ManuallyDrop	0
R21 RULE - Always call from_raw on into_rawed value	0
R22 RULE - Do not use uninitialized memory	0
R32 RULE - Use only C-compatible types in FFI	0

무기체계 소프트웨어 보안약점 점검 목록

Reference Chapter	Issues
CWE-119	0
CWE-134	0
CWE-170	0
CWE-190	1
CWE-209	53
CWE-22	9
CWE-259	0
CWE-285	0
CWE-306	0
CWE-307	0
CWE-312	0
CWE-319	0
CWE-321	0
CWE-327	7
CWE-330	4
CWE-367	1
CWE-369	0
CWE-390	6
CWE-400	0
CWE-404	23
CWE-415	0
CWE-416	0
CWE-457	0
CWE-467	0
CWE-469	0
CWE-476	34
CWE-489	0
CWE-494	0
CWE-495	0
CWE-496	0
CWE-497	53
CWE-521	0
CWE-562	0
CWE-587	0

CWE-59	0
CWE-615	0
CWE-628	0
CWE-676	0
CWE-732	0
CWE-755	20
CWE-759	0
CWE-78	5
CWE-89	2
CWE-99	9

● 방위사업청 코딩규칙

Reference Chapter	Issues
1-01. Switch 구문에서 첫 번째 Label 전에 코드 구문이 존재하면 안된다.	0
1-02. 함수/변수 선언 시 type을 명시해야 한다.	0
1-03. 의미 없는 구문은 사용하지 말아야 한다.(side effect)	0
1-04. 함수의 Return Type에 맞는 return을 사용해야 한다.	0
1-05. 선언 없이 함수를 사용하지 말아야 한다.(묵시적 선언이 사용됨)	0
1-06. 매크로의 정의 여부를 확인하지 않고 해당 매크로에 대하여 #if, #elseif 표현을 사용하지 말아야 한다.	0
1-07. goto 문 사용은 최대한 자제한다.	0
1-08. 하나의 함수는 하나의 Exit Point를 가져야 한다.	0
1-09. switch~case 문은 default 문이 포함되어야 한다.	0
1-10. 한 줄에 하나의 명령문을 사용한다.	0
1-11. if - else if 문은 else 문도 포함시킨다.	0
2-01. String 배열의 초기화에서 배열의 마지막 인자는 NULL로 종료되어야 한다.	0
2-02. 초기화 되지 않은 변수를 사용하지 말아야 한다.	0
2-03. 설정되지 않은 포인터를 함수의 Read-only(const)로 사용하면 안된다.	0
3-01. external과 internal linkage 의 특성을 동시에 가질 수 없다.	0
3-02. external linkage scope 에서 선언된 함수나 Object의 이름은 유일해야 한다.	0
3-03. external linkage scope 에서 정의된 함수나 Object의 데이터 형은 선언 시 정의와 동일해야 한다.	0
3-04. 바깥 scope 의 식별자를 가리는 정의를 해서는 안된다.	0
4-01. float 자료형에서 동등성 비교연산을 수행하지 말아야 한다.	0
4-02. 조건문의 결과가 항상 True거나 False면 안된다.	0

4-03. switch의 case 조건을 만족할 수 없는 Label을 사용하지 않는다.	0
4-04. switch 구문에서 Expression을 논리적 연산으로 사용하지 말아야 한다.	0
4-05. 수행되지 않는 소스코드를 작성하지 말아야 한다.	0
5-01. 선언된 데이터 형으로 표현할 수 있는 숫자의 영역을 초과하는 값을 할당하지 말아야 한다.	0
5-02. 가변인수를 받는 함수의 Conversion 지시자와 Argument의 type은 동일해야 한다.	0
5-03. 가변인수를 받는 함수의 Conversion 지시자와 Argument의 개수는 동일해야 한다.	0
5-04. Object 저장값을 표현할 수 없는 데이터로의 형 변환을 하지말아야 한다.	0
5-05. 음수값을 unsigned type으로 변환을 자제해야 한다.	0
5-06. Character 문자열과 Wide character 문자열을 혼용하지 말아야 한다.	0
5-07. 포인터 Cast의 결과로 이전 포인터의 Const 특성의 상실을 유의해야 한다.	0
5-08. 포인터 Cast의 결과로 이전 포인터의 Volatile 특성의 상실을 유의해야 한다.	0
6-01. Null pointer를 참조하지 않는다.	0
6-02. 지역 변수의 주소값을 더 넓은 scope를 가진 변수에 할당하지 말아야 한다.	0
6-03. 지역 변수의 주소값을 함수의 리턴값으로 사용하지 말아야 한다.	0
6-04. 선언된 배열의 크기를 초과하는 인덱스 값을 사용하지 말아야 한다.	0
6-05. Null Pointer를 산술연산 하지 않는다.	0
7-01. 하나의 Sequence Point 내에서 하나의 Object Value를 두 번 이상 변경하지 않아야 한다.	0
7-02. 0 으로 나눗셈 연산을 하지 않는다.	0
7-03. 하나의 Sequence Point 내에서 Object의 값을 변경하고 Access 하지 않아야 한다.	0
7-04. 음수 값 또는 데이터 사이즈를 초과하는 값을 사용하여 Shift operator를 하지 않는다.	0
7-05. Underlying type이 부호 없는 정수일 경우 단행 빼기 연산(-)을 사용하여 결과를 대입하지 말아야 한다.	0
7-06. sizeof의 인자는 side effect를 가지지 말아야 한다.	0
7-07. Boolean 표현 값에 &&, ||, ! 연산자를 제외하고 다른 연산자를 사용하지 말아야 한다.	0
7-08. 조건문에 직접적인 대입 연산자를 사용하지 말아야 한다.	0
7-09. Signed Value에서 Bitwise연산자(<<, ~, |, ^ 등)로 인한 Negative Value를 유의해야 한다.	0
8-01. Scanf의 Argument 는 Object Value의 저장된 주소에 값이 입력되어야 한다.	0
8-02. #include 구문에서 표준에 맞지 않는 Character set을 사용하지 않아야 한다.	0
8-03. Allocated되는 메모리 블록의 크기는 Pointer에 의해서 Address 되는 완전한 하나의 multiple size여야 한다.	0
8-04. 함수의 Argument type과 개수는 함수의 Prototype, 선언, 정의가 모두 같아야 한다.	0
8-05. 구조체/배열의 초기화 시 default 초기화 값(0)을 제외하고, 구조에 맞게 ‘{}’를 사용하여 선언된 Size에 맞게 초기화 해야 한다.	0

9-01. 동적 할당된 데이터를 해제할 때, 잘못된 메소드를 이용하여 해제하면 안된다.	0
9-02. 지역 변수의 주소 값을 처리하는 handle을 return하지 말아야 한다.	0
9-03. 함수 parameter의 주소 값을 처리하는 handle을 return하지 말아야 한다.	0
9-04. 소멸자내에서 처리할 수 없는 예외 상황을 발생시키지 말아야 한다.	0
9-05. 사용되지 않는 예외 처리 문을 작성하지 말아야 한다.	0
9-06. exception specification에 기술되지 않은 모든 throw에 대하여 예외처리를 해야만 한다.	0
9-07. main 함수에서 처리되지 않는 throw를 작성하지 말아야 한다.	0
9-08. 해제된 메모리 영역 사용하지 말아야 한다.	0
9-09.복사 연산자를 통해서, 복사되지 않는 멤버 변수가 존재하지 말아야 한다	0
9-10. C 코딩 방법으로 메모리를 할당 하면 안된다.	0
9-11. 순수 가상함수는 반드시 0으로 초기화 되어야 한다	0
9-12. 순수함수는 반드시 가상함수로 선언되어야 한다	0
9-13. virtual base 클래스의 포인터는 derived 클래스의 포인터로 cast 할 때에는 dynamic_cast만 사용해야 한다.	0
9-14. 생성자/소멸자 내에서 가상함수는 식별자 없이 호출하면 안된다.	0
9-15. 생성자/소멸자에 dynamic type을 사용하면 안된다.	0

소프트웨어 보안약점 진단가이드 2021

Reference Chapter	Issues
DNS lookup에 의존한 보안 결정	0
HTTP 응답분할	7
LDAP 삽입	3
Null Pointer 역참조	34
Private 배열에 Public 데이터 할당	0
Public 메소드부터 반환된 Private 배열	0
SQL 삽입	2
XML 삽입	2
경로 조작 및 자원 삽입	18
경쟁조건: 검사시점과 사용시점(TOCTOU)	1
메모리 버퍼 오버플로우	0
무결성 검사없는 코드 다운로드	0
반복된 인증시도 제한 기능 부재	0
보안기능 결정에 사용되는 부적절한 입력값	26
부적절한 XML 외부개체 참조	0

부적절한 예외처리	20
부적절한 인가	0
부적절한 인증서 유효성 검증	0
부적절한 자원 해제	23
부적절한 전자서명 확인	0
사용자 하드디스크에 저장되는 쿠키를 통한 정보 노출	0
서버사이드 요청 위조	0
솔트 없이 일방향 해쉬 함수 사용	0
신뢰되지 않는 URL 주소로 자동 접속 연결	0
신뢰할 수 없는 데이터의 역직렬화	0
암호화되지 않은 중요정보	2
오류 상황 대응 부재	6
오류메시지 정보 노출	53
운영체제 명령어 삽입	5
위험한 형식 파일 업로드	0
잘못된 세션에 의한 데이터 정보 노출	0
적절하지 않은 난수 값 사용	4
적절한 인증없는 중요기능 허용	0
정수형 오버플로우	1
제거되지 않고 남은 디버그 코드	0
종료되지 않는 반복문 또는 재귀 함수	0
주석문 안에 포함된 시스템 주요정보	0
중요한 자원에 대한 잘못된 권한 설정	0
초기화되지 않은 변수 사용	0
충분하지 않은 키 길이 사용	0
취약한 API 사용	0
취약한 비밀번호 허용	0
취약한 암호화 알고리즘 사용	7
코드 삽입	0
크로스사이트 스크립트	22
크로스사이트 요청 위조	0
포맷스트링 삽입	0
하드코드된 중요정보	0
해제된 자원 사용	0

● 주요정보통신기반시설 취약점 분석·평가 기준

Reference Chapter	Issues
SQL 인젝션	0
XPath 인젝션	0
경로 추적	0
디렉토리 인덱싱	0
세션 고정	0
세션 예측	0
약한 문자열 강도	0
운영체제 명령 실행	0
위치 공개	0
크로스사이트 스크립팅	0
파일 다운로드	0

Issue Details

[Rule Name] return Statement in catch Block (Medium, Java)

The return Statement in catch Block checker finds catch blocks that contain a return statement.

Store the return value in a variable and have it returned after the end of the whole try block.

- CWE 660 4.14
  - 460 - Improper Cleanup on Thrown Exception

- CWE 660 4.7
  - Improper Cleanup on Thrown Exception - (460)

Dangerous Example

```
public static final boolean doStuff( ) {

  boolean threadLock;
  boolean truthvalue=true;
  try {

  while(
//check some condition
  ) {

      threadLock=true; //do some stuff to truthvalue
      threadLock=false;
      }
    }
    catch (Exception e){

    System.err.println("You did something bad");
    if (something) return truthvalue;
19.  }
20.  return truthvalue;
21. }
```

Line 18: A value is returned in a catch block.

Safe Example

```
1. public static final boolean doStuff( ) {
2.
3.  boolean threadLock;
4.  boolean truthvalue=true;
5.  try {
6.
7.  while(
8. //check some condition
9.  ) {
10.
11.    threadLock=true; //do some stuff to truthvalue
12.    threadLock=false;
13.    }
14.  }
15.  catch (Exception e){
16.
17.  System.err.println("You did something bad");
18.  //if (something) return truthvalue;
19.  }
20.  return truthvalue;
21. }
```

Line 18: Do not use the return statements in the catch block.

Issue ID	274457
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DataBaseServer.java
Line	72

Source Code

```
67.             org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, resp);
68.         } catch (java.sql.SQLException e) {
69.             if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
70.                 e.printStackTrace();
71.                 resp.add(new XMLMessage("Error processing request: " + e.getMessage()));
72.                 return new ResponseEntity<List<XMLMessage>>(resp, HttpStatus.OK);
73.             } else throw new ServletException(e);
74.         }
75.         return new ResponseEntity<List<XMLMessage>>(resp, HttpStatus.OK);
76.     }
77.
```

Issue ID	274472
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /LDAPManager.java
Line	131

Source Code

```
126.
127.            return true;
128.        } catch (Exception e) {
129.            System.out.println("LDAP error search: ");
130.            e.printStackTrace();
131.            return false;
132.        }
133.    }
134.
135.    public DirContext getDirContext() throws NamingException {
136.        if (ctx == null) {
```

Issue ID	274498

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00007.java

Line 67

Source Code

```
62.            org.owasp.benchmark.helpers.Utils.printOSCommandResults(p, response);
63.        } catch (IOException e) {
64.            System.out.println("Problem executing cmdi - TestCase");
65.            response.getWriter()
66.                    .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(e.
getMessage()));
67.            return;
68.        }
69.    }
70. }
```

Issue ID 274500

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00017.java

Line 69

Source Code

```
64.            org.owasp.benchmark.helpers.Utils.printOSCommandResults(p, response);
65.        } catch (IOException e) {
66.            System.out.println("Problem executing cmdi - TestCase");
67.            response.getWriter()
68.                    .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(e.
getMessage()));
69.            return;
70.        }
71.    }
72. }
```

Issue ID 274501

File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00018.java
Line	64

Source Code

```
59.            int count = statement.executeUpdate(sql);
60.            org.owasp.benchmark.helpers.DatabaseHelper.outputUpdateComplete(sql, response);
61.        } catch (java.sql.SQLException e) {
62.            if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
63.                response.getWriter().println("Error processing request.");
64.                return;
65.            } else throw new ServletException(e);
66.        }
67.    }
68. }
```

Issue ID	274514
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00034.java
Line	61

Source Code

```
56.            statement.execute(sql, java.sql.Statement.RETURN_GENERATED_KEYS);
57.            org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, response);
58.        } catch (java.sql.SQLException e) {
59.            if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
60.                response.getWriter().println("Error processing request.");
61.                return;
62.            } else throw new ServletException(e);
63.        }
64.    }
65. }
```

Issue ID	274536
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00092.java
Line	101

Source Code

```
96.             org.owasp.benchmark.helpers.Utils.printOSCommandResults(p, response);
97.         } catch (IOException e) {
98.             System.out.println("Problem executing cmdi - TestCase");
99.             response.getWriter()
100.                    .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(e.getMessage()));
101.            return;
102.        }
103.    }
104. }
```

● [Rule Name] Race Condition for Database Connection (Medium, Java)

The Race Condition for Database Connection checker finds static declarations of database connection objects.

A java.sql.Connection object is regarded as a database connection object.

Transactional resource objects, such as JDBC connections, must not be stored in static fields. Such an object can only be associated with one transaction at a time. For this reason, storing it in a static field causes invalid sharing between different transactional threads. Therefore, do not store database connections in static fields. In a multithreaded program, if multiple threads concurrently access a transactional resource stored in a static field, a race condition may occur leading to program malfunctions whose causes are not easy to find.

Transactional resource objects are recommended to be stored in instance fields.

- CWE 660 4.14

    - 362 - Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')

    - 366 - Race Condition within a Thread

- CWE 660 4.7

    - Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition') - (362)

Dangerous Example

```
1. public class DBConnectionManager {
2.   private static Connection conn = connect(); //Avoid static
3. }
```

Line 2: A database connection object is stored in static field.

Safe Example

```
1. public class DBConnectionManager {
2.   private Connection conn = connect();
3. }
```

Line 2: Do not store the database connection object in static field.

Issue ID	274461
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DatabaseHelper.java
Line	36

Source Code

```
31. import javax.sql.DataSource;
32. import org.owasp.benchmark.service.pojo.XMLMessage;
33. import org.owasp.esapi.ESAPI;
```

```

public class DatabaseHelper {
    private static Connection conn;
    public static org.springframework.jdbc.core.JdbcTemplate JDBCtemplate;
    public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtil =
            new org.owasp.benchmark.helpers.HibernateUtil(false);
    public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtilClassic =
            new org.owasp.benchmark.helpers.HibernateUtil(true);
```

● [Rule Name] Resource leak (Medium, Java)

The Resource leak checker finds instances of failure to release an allocated resource (file, socket, etc.).

If a finite program resource is once allocated and no longer used, it must be reclaimed. If a program error or the JVM garbage collector hinders resources from being quickly reclaimed, the program may run out of resources. This may lead to system performance degradation, interruption of functionality, denial of service (DoS), or failure to claim another resource.

Make sure any resource allocated in a method is freed before the same method ends. Because such a resource often causes an exception when accessed, it is recommended to enclose it with a try block and have possible exceptions thrown in a finally block.

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-404

- 소프트웨어 보안약점 진단가이드 2021
  - 부적절한 자원 해제

Dangerous Example

```
public class ResourceLeakEx {
  public void testResourceLeak() {
    try {
      BufferedWriter out = new BufferedWriter(new FileWriter(
                                                          new File("test.txt")));
6.       out.write("This is Resource Leak sample code...");
7.       out.newLine();
8.     } catch (IOException e) {
9.       // ...
10.    }
11.  } /* BUG : RESOURCE_LEAK */
12. }
```

Line 4: When an exception occurs after resource allocation, its resource is not released.

Safe Example

```
1. public class ResourceLeakSafeEx {
2.   public void testResourceLeak() {
3.     try {
4.       BufferedWriter out = new BufferedWriter(new FileWriter(
5.                                                       new File("test.txt")));
6.       out.write("This is Resource Leak sample code...");
7.       out.newLine();
8.     } catch (IOException e) {
9.       // ...
10.    } finally {
11.      if (out != null) {
12.        out.close(); /* SAFE */
13.      }
14.    }
15.  }
16. }
```

Line 12: Free the resource without considering exceptions.

Issue ID	274707
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DataBaseServer.java
Line	65

Source Code

```
60.     List<XMLMessage> resp = new ArrayList<XMLMessage>();
61.     String sql = "SELECT * from USERS";
62.     try {
63.         java.sql.Connection connection =
64.                 org.owasp.benchmark.helpers.DatabaseHelper.getSqlConnection();
65.         java.sql.PreparedStatement statement = connection.prepareStatement(sql);
66.         statement.execute();
67.         org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, resp);
68.     } catch (java.sql.SQLException e) {
69.         if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
70.             e.printStackTrace();
```

Issue ID	274538
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DataBaseServer.java
Line	65

Source Code

```
60.     List<XMLMessage> resp = new ArrayList<XMLMessage>();
61.     String sql = "SELECT * from USERS";
62.     try {
63.         java.sql.Connection connection =
64.                 org.owasp.benchmark.helpers.DatabaseHelper.getSqlConnection();
65.         java.sql.PreparedStatement statement = connection.prepareStatement(sql);
66.         statement.execute();
67.         org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, resp);
68.     } catch (java.sql.SQLException e) {
69.         if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
70.             e.printStackTrace();
```

Issue ID	274540
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers

File /DatabaseHelper.java

Line 124

Source Code

```
119.        if (conn == null) {
120.            getSqlConnection();
121.        }
122.        Statement stmt = null;
123.        try {
124.            stmt = conn.createStatement();
125.        } catch (SQLException e) {
126.            System.out.println("Problem with database init.");
127.        }
128.
129.        return stmt;
```

Issue ID 274594

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DatabaseHelper.java

Line 124

Source Code

```
119.        if (conn == null) {
120.            getSqlConnection();
121.        }
122.        Statement stmt = null;
123.        try {
124.            stmt = conn.createStatement();
125.        } catch (SQLException e) {
126.            System.out.println("Problem with database init.");
127.        }
128.
129.        return stmt;
```

Issue ID	274611
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DatabaseHelper.java
Line	124

Source Code

```
119.        if (conn == null) {
120.            getSqlConnection();
121.        }
122.        Statement stmt = null;
123.        try {
124.            stmt = conn.createStatement();
125.        } catch (SQLException e) {
126.            System.out.println("Problem with database init.");
127.        }
128.
129.        return stmt;
```

Issue ID	274541
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DatabaseHelper.java
Line	161

Source Code

```
156.    public static java.sql.Connection getSqlConnection() {
157.        if (conn == null) {
158.            try {
159.                InitialContext ctx = new InitialContext();
160.                DataSource datasource = (DataSource) ctx.lookup("java:comp/env/jdbc/BenchmarkDB");
161.                conn = datasource.getConnection();
162.                conn.setAutoCommit(false);
163.            } catch (SQLException | NamingException e) {
```

```
164.            System.out.println("Problem with getSqlConnection.");
165.            e.printStackTrace();
166.        }
```

Issue ID	274542
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DatabaseHelper.java
Line	161

Source Code

```
156.    public static java.sql.Connection getSqlConnection() {
157.        if (conn == null) {
158.            try {
159.                InitialContext ctx = new InitialContext();
160.                DataSource datasource = (DataSource) ctx.lookup("java:comp/env/jdbc/BenchmarkDB");
161.                conn = datasource.getConnection();
162.                conn.setAutoCommit(false);
163.            } catch (SQLException | NamingException e) {
164.                System.out.println("Problem with getSqlConnection.");
165.                e.printStackTrace();
166.            }
```

Issue ID	274557
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	234

Source Code

```
229.                    + "<meta http-equiv=₩"Content-Type₩" content=₩"text/html; charset=UTF-8₩">₩n"
230.                    + "</head>₩n"
231.                    + "<body>₩n"
```

```
232.                    + "<p>₩n");
233.
234.        BufferedReader stdInput = new BufferedReader(new InputStreamReader
(proc.getInputStream()));
235.        BufferedReader stdError = new BufferedReader(new InputStreamReader
(proc.getErrorStream()));
236.
237.        try {
238.            // read the output from the command
239.            // System.out.println("Here is the standard output of the
```

Issue ID	274555
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	235

Source Code

```
230.                    + "</head>₩n"
231.                    + "<body>₩n"
232.                    + "<p>₩n");
233.
234.        BufferedReader stdInput = new BufferedReader(new InputStreamReader
(proc.getInputStream()));
235.        BufferedReader stdError = new BufferedReader(new InputStreamReader
(proc.getErrorStream()));
236.
237.        try {
238.            // read the output from the command
239.            // System.out.println("Here is the standard output of the
240.            // command:₩n");
```

Issue ID	274556
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java

Line 235

Source Code

```
230.                    + "</head>₩n"
231.                    + "<body>₩n"
232.                    + "<p>₩n");
233.
234.         BufferedReader stdInput = new BufferedReader(new InputStreamReader(proc.getInputStream()));
235.         BufferedReader stdError = new BufferedReader(new InputStreamReader(proc.getErrorStream()));
236.
237.         try {
238.             // read the output from the command
239.             // System.out.println("Here is the standard output of the
240.             // command:₩n");
```

Issue ID	274552
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	266

Source Code

```
261.
262.     // A method used by the Benchmark JAVA test cases to format OS Command Output
263.     // This version is only used by the Web Services test cases.
264.     public static void printOSCommandResults(java.lang.Process proc, List<XMLMessage> resp) {
265.
266.         BufferedReader stdInput = new BufferedReader(new InputStreamReader(proc.getInputStream()));
267.         BufferedReader stdError = new BufferedReader(new InputStreamReader(proc.getErrorStream()));
268.
269.         try {
```

```
270.            // read the output from the command
271.            resp.add(new XMLMessage("Here is the standard output of the command:"));
```

Issue ID	274553
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	267

Source Code

```
262.    // A method used by the Benchmark JAVA test cases to format OS Command Output
263.    // This version is only used by the Web Services test cases.
264.    public static void printOSCommandResults(java.lang.Process proc, List<XMLMessage> resp) {
265.
266.        BufferedReader stdInput = new BufferedReader(new InputStreamReader(proc.getInputStream()));
267.        BufferedReader stdError = new BufferedReader(new InputStreamReader(proc.getErrorStream()));
268.
269.        try {
270.            // read the output from the command
271.            resp.add(new XMLMessage("Here is the standard output of the command:"));
272.            String s = null;
```

Issue ID	274554
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	267

Source Code

```
262.    // A method used by the Benchmark JAVA test cases to format OS Command Output
263.    // This version is only used by the Web Services test cases.
264.    public static void printOSCommandResults(java.lang.Process proc, List<XMLMessage> resp) {
265.
266.        BufferedReader stdInput = new BufferedReader(new InputStreamReader(proc.getInputStream()));
267.        BufferedReader stdError = new BufferedReader(new InputStreamReader(proc.getErrorStream()));
268.
269.        try {
270.            // read the output from the command
271.            resp.add(new XMLMessage("Here is the standard output of the command:"));
272.            String s = null;
```

Issue ID	274558
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	383

Source Code

```
378.            Files.createDirectories(pathToFileDir);
379.            File f = new File(completeName);
380.            if (!f.exists()) {
381.                f.createNewFile();
382.            }
383.            FileOutputStream fos = new FileOutputStream(f, true);
384.            os = new PrintStream(fos);
385.            os.println(line);
386.        } catch (IOException e1) {
387.            result = false;
388.            e1.printStackTrace();
```

Issue ID	274577
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00003.java
Line	94

Source Code

```
89.         java.io.File fileTarget =
90.                 new java.io.File(
91.                         new java.io.File(org.owasp.benchmark.helpers.Utils.TESTFILES_DIR),
92.                         "passwordFile.txt");
93.         java.io.FileWriter fw =
94.                 new java.io.FileWriter(fileTarget, true); // the true will append the new data
95.         fw.write(
96.                 "hash_value="
97.                         + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
98.                         + "₩n");
99.         fw.close();
```

Issue ID	274578
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00005.java
Line	93

Source Code

```
88.         java.io.File fileTarget =
89.                 new java.io.File(
90.                         new java.io.File(org.owasp.benchmark.helpers.Utils.TESTFILES_DIR),
91.                         "passwordFile.txt");
92.         java.io.FileWriter fw =
93.                 new java.io.FileWriter(fileTarget, true); // the true will append the new data
94.         fw.write(
95.              "secret_value="
96.                   + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
97.                   + "₩n");
98.         fw.close();
```

Issue ID	274584
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00009.java
Line	94

Source Code

```
89.         java.io.File fileTarget =
90.              new java.io.File(
91.                   new java.io.File(org.owasp.benchmark.helpers.Utils.
TESTFILES_DIR),
92.                   "passwordFile.txt");
93.         java.io.FileWriter fw =
94.              new java.io.FileWriter(fileTarget, true); // the true will append the
new data
95.         fw.write(
96.              "hash_value="
97.                   + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
98.                   + "₩n");
99.         fw.close();
```

Issue ID	274596
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00020.java
Line	88

Source Code

```
83.         java.io.File fileTarget =
84.              new java.io.File(
85.                     new java.io.File(org.owasp.benchmark.helpers.Utils.TESTFILES_DIR),
86.                     "passwordFile.txt");
87.           java.io.FileWriter fw =
88.                new java.io.FileWriter(fileTarget, true); // the true will append the new data
89.           fw.write(
90.                "secret_value="
91.                     + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
92.                     + "₩n");
93.           fw.close();
```

Issue ID	274624
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00050.java
Line	89

Source Code

```
84.           java.io.File fileTarget =
85.                new java.io.File(
86.                     new java.io.File(org.owasp.benchmark.helpers.Utils.TESTFILES_DIR),
87.                     "passwordFile.txt");
88.           java.io.FileWriter fw =
89.                new java.io.FileWriter(fileTarget, true); // the true will append the new data
90.           fw.write(
91.                "secret_value="
92.                     + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
93.                     + "₩n");
94.           fw.close();
```

Issue ID	274659
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode

File	/BenchmarkTest00070.java
Line	96

Source Code

```
91.          java.io.File fileTarget =
92.                  new java.io.File(
93.                          new java.io.File(org.owasp.benchmark.helpers.Utils.TESTFILES_DIR),
94.                          "passwordFile.txt");
95.          java.io.FileWriter fw =
96.                  new java.io.FileWriter(fileTarget, true); // the true will append the new data
97.          fw.write(
98.                  "hash_value="
99.                          + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
100.                         + "₩n");
101.         fw.close();
```

Issue ID	274666
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00076.java
Line	114

Source Code

```
109.         java.io.File fileTarget =
110.                 new java.io.File(
111.                         new java.io.File(org.owasp.benchmark.helpers.Utils.TESTFILES_DIR),
112.                         "passwordFile.txt");
113.         java.io.FileWriter fw =
114.                 new java.io.FileWriter(fileTarget, true); // the true will append the new data
115.         fw.write(
116.                 "hash_value="
117.                         + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result,
true)
118.                          + "₩n");
119.          fw.close();
```

Issue ID 274700

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00207.java

Line 62

Source Code

```
57.                                         param.getBytes())));
58.     }
59.
60.     try {
61.         java.io.FileInputStream file =
62.                 new java.io.FileInputStream(
63.                         org.owasp.benchmark.helpers.Utils.getFileFromClasspath(
64.                                 "employees.xml", this.getClass().getClassLoader()));
65.         javax.xml.parsers.DocumentBuilderFactory builderFactory =
66.                 javax.xml.parsers.DocumentBuilderFactory.newInstance();
67.         // Prevent XXE
```

Issue ID 274705

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00442.java

Line 55

Source Code

```
50.
51.     bar = (7 * 42) - num > 200 ? "This should never happen" : param;
52.
53.     try {
54.         java.io.FileInputStream file =
            new java.io.FileInputStream(
                org.owasp.benchmark.helpers.Utils.getFileFromClasspath(
                    "employees.xml", this.getClass().getClassLoader()));
        javax.xml.parsers.DocumentBuilderFactory builderFactory =
            javax.xml.parsers.DocumentBuilderFactory.newInstance();
        // Prevent XXE
```

● [Rule Name] Null dereference (Medium, Java)

The Null dereference checker finds instances of dereferencing a null constant or a null-assigned variable without checking it for null.

If a value that might be null is dereferenced, a NullPointerException exception may occur during program execution. This may result in an abnormal termination of the program.

Attackers can use vulnerabilities caused by the NullPointerException exception to plan later attacks.

Unless a variable is ensured to never be null, make sure it is always checked for null before dereferenced.

- CWE 660 4.14
    - 476 - NULL Pointer Dereference

- CWE 660 4.7
    - NULL Pointer Dereference - (476)

- 무기체계 소프트웨어 보안약점 점검 목록
    - CWE-476

- 소프트웨어 보안약점 진단가이드 2021
    - Null Pointer 역참조

Dangerous Example

```
1. public class ForwardNullEx {
2.   public void test() {
3.     String uppercased = toUpperCase(null); // FORWARD_NULL
4.   }
5.   public String toUpperCase(String arg) {
6.     arg.toUpperCase();
7.   }
8. }
```

Line 3: A null value is passed to toUpperCase() method.
Line 6: An argument is dereferenced without a null check.

Safe Example

```
1. public class ForwardNullSafeEx {
2.   public void test() {
3.     String uppercased = toUpperCase(null);  // SAFE
4.   }
5.   public String toUpperCase(String arg) {
6.     if (arg != null) { // Do a null check
7.       arg.toUpperCase();
8.     } else {
9.       return null;
10.     }
11.   }
12. }
```

Line 6: Use variables after a null check to avoid a null dereference.

Issue ID	274559
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	390

Source Code

```
            os.println(line);
        } catch (IOException e1) {
            result = false;
            e1.printStackTrace();
        } finally {
            os.close();
        }

        return result;
    }

```

[Rule Name] Null return value dereference (Medium, Java)

The Null return value dereference checker finds instances of dereferencing a return value without checking it for null.

If a method''s return is dereferenced without a null check, an exception may be thrown.

A NullPointerException exception allows for a vulnerability exploited to plan attacks.

A return value from a method that can return null must be checked for null before used.

- CWE 660 4.14
    - 476 - NULL Pointer Dereference
- CWE 660 4.7
    - NULL Pointer Dereference - (476)
- 무기체계 소프트웨어 보안약점 점검 목록
    - CWE-476
- 소프트웨어 보안약점 진단가이드 2021
    - Null Pointer 역참조

Dangerous Example

```
1. public class NullReturnEx {
2.   public Object returnNull() { return null; }
3.   public void testNull() {
4.     String str =returnNull().toString(); /* BUG : NULL_RETURN */
5.   }
6. }
```

Line 2: returnNull() method returns a null.
Line 4: A return value of returnNull() method is dereferenced without a null check.

Safe Example

```
1. public class NullReturnSafeEx {
2.   public Object returnNull() { return null; }
3.   public void testNull() {
4.     Object x = returnNull();
5.     String str = x != null ? x.toString() : null;  // SAFE
6.   }
7. }
```

Line 5: When calling a method that returns null, check a null before using its return value.

Issue ID	274545
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /LDAPServer.java
Line	96

Source Code

```
91.        // BEGIN HACK
92.        try {
93.            String dir =
94.                    Utils.getFileFromClasspath(
95.                                    "benchmark.properties", LDAPServer.class.getClassLoader())
96.                            .getParent();
```

```
97.         File workDir = new File(dir + "/../ldap");
98.         workDir.mkdirs();
99.         System.setProperty("workingDiretory", workDir.getPath());
100.
101.          init();
```

Issue ID	274546
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	153

Source Code

```
148.         perms.add(PosixFilePermission.GROUP_EXECUTE);
149.         perms.add(PosixFilePermission.OTHERS_READ);
150.         perms.add(PosixFilePermission.OTHERS_EXECUTE);
151.
152.         try {
153.             Files.setPosixFilePermissions(script.toPath(), perms);
154.         } catch (IOException e) {
155.             System.out.println(
156.                     "Problem while changing executable permissions: " + e.
getMessage());
157.         }
158.     }
```

Issue ID	274550
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	192

Source Code

```
187.
188.     public static String getInsecureOSCommandString(ClassLoader classLoader) {
```

```
189.        String command = null;
190.        String osName = System.getProperty("os.name");
191.        if (osName.indexOf("Windows") != -1) {
192.            command = Utils.getFileFromClasspath("insecureCmd.bat", classLoader).getAbsolutePath();
193.        } else {
194.            command = Utils.getFileFromClasspath("insecureCmd.sh", classLoader).getAbsolutePath();
195.        }
196.        return command;
197.    }
```

Issue ID	274549
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	194

Source Code

```
189.        String command = null;
190.        String osName = System.getProperty("os.name");
191.        if (osName.indexOf("Windows") != -1) {
192.            command = Utils.getFileFromClasspath("insecureCmd.bat", classLoader).getAbsolutePath();
193.        } else {
194.            command = Utils.getFileFromClasspath("insecureCmd.sh", classLoader).getAbsolutePath();
195.        }
196.        return command;
197.    }
198.
199.    public static List<String> getOSCommandArray(String append) {
```

● [Rule Name] Missing null check (Medium, Java)

The Missing null check checker finds instances in which a variable is once checked for null but later is dereferenced without a null check.

If a variable is checked for null at least once, it implies that when writing the null check code, the programmer considered the variable might get null through the execution path reaching that point. And if the same variable is later dereferenced without null check, it is highly likely that the programmer has made a mistake. A NullPointerException exception allows for a vulnerability exploited to plan attacks.

Unless a variable is ensured to never be null, make sure it is always checked for null before dereferenced. Conversely, if a variable can never be null and so its null check is unnecessary, have it never checked for null, ensuring consistency and preventing confusion.

- CWE 660 4.14
  - 476 - NULL Pointer Dereference

- CWE 660 4.7
  - NULL Pointer Dereference - (476)

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-476

- 소프트웨어 보안약점 진단가이드 2021
  - Null Pointer 역참조

Dangerous Example

```
public class UncheckedNullEx {
  public void test(String x) {
    String str = "";
    System.out.println(str);
    if(x != null) {
      str = x.toUpperCase();
    }
8.    x.toString(); /* BUG : UNCHECKED_NULL */
9.  }
10. }
```

Line 5: Variable x is compared to null.
Line 8: The variable x, that was compared to null, is used without check.

Safe Example

```
1. public class UncheckedNullSafeEx {
2.   public void test(String x) {
3.     String str = "";
4.     System.out.println(str);
5.     if(x != null) {
6.       str = x.toUpperCase();
7.     }
8.     if(x != null) {
9.       x.toString(); /* SAFE */
10.     }
11.   }
12. }
```

Line 9: A variable is dereferenced after a null check.

Issue ID	274706
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DataBaseServer.java
Line	65

Source Code

```
60.        List<XMLMessage> resp = new ArrayList<XMLMessage>();
61.        String sql = "SELECT * from USERS";
62.        try {
63.            java.sql.Connection connection =
64.                    org.owasp.benchmark.helpers.DatabaseHelper.getSqlConnection();
65.            java.sql.PreparedStatement statement = connection.prepareStatement(sql);
66.            statement.execute();
```

```
67.            org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, resp);
68.        } catch (java.sql.SQLException e) {
69.            if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
70.                e.printStackTrace();
```

Issue ID	274537
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DataBaseServer.java
Line	65

Source Code

```
60.        List<XMLMessage> resp = new ArrayList<XMLMessage>();
61.        String sql = "SELECT * from USERS";
62.        try {
63.            java.sql.Connection connection =
64.                    org.owasp.benchmark.helpers.DatabaseHelper.getSqlConnection();
65.            java.sql.PreparedStatement statement = connection.prepareStatement(sql);
66.            statement.execute();
67.            org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, resp);
68.        } catch (java.sql.SQLException e) {
69.            if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
70.                e.printStackTrace();
```

Issue ID	274539
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java
Line	124

Source Code

```
119.        if (conn == null) {
120.            getSqlConnection();
```

```
    }
    Statement stmt = null;
    try {
        stmt = conn.createStatement();
    } catch (SQLException e) {
        System.out.println("Problem with database init.");
    }

    return stmt;
```

● [Rule Name] Null return value dereference in standard library (Medium, Java)

The Null return value dereference in standard library checker finds instances of dereferencing a return value from a Java standard library method without checking it for null even through the method can return null.

If a standard library method''s return is dereferenced without a null check, an exception may be thrown.

A NullPointerException exception allows for a vulnerability exploited to plan attacks.

A return value from a standard library method that can return null must be checked for null before used.

- CWE 660 4.14
    - 476 - NULL Pointer Dereference

- CWE 660 4.7
    - NULL Pointer Dereference - (476)

- 무기체계 소프트웨어 보안약점 점검 목록
    - CWE-476

- 소프트웨어 보안약점 진단가이드 2021

- Null Pointer 역참조

Dangerous Example

```
1. public class NullReturnStdEx {
2.   public void getInputFromFile() {
3.     try  {
4.       BufferedReader br =
5.         new BufferedReader(new FileReader("input.dat"));
6.       String str = br.readLine();  // BufferedReader.readLine() can return null
7.       str.toUpperCase();  // NULL_RETURN_STD
8.     } catch (IOException e) { e.printStackTrace(); }
9.   }
10. }
```

Line 6: A return value of readline() method of java.io.BufferedReader class can be a null. If a null is returned, a null dereference occurs by calling toUpperCase() method.

Safe Example

```
1. public class NullReturnStdSafeEx {
2.   public void getInputFromFile() {
3.     try  {
4.       BufferedReader br =
5.         new BufferedReader(new FileReader("input.dat"));
6.       String str = br.readLine();  // BufferedReader.readLine() can return null
7.       if (str != NULL) {
8.         str.toUpperCase();  // SAFE
9.       }
10.     } catch (IOException e) { e.printStackTrace(); }
11.   }
12. }
```

Line 7: Use the return value of readLine() method after a null check.

Issue ID	274543
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers

File	/LDAPManager.java
Line	84

Source Code

```
79.         InitialDirContext iniDirContext = (InitialDirContext) ctx;
80.
81.         try {
82.             iniDirContext.bind(name, ctx, matchAttrs);
83.         } catch (NamingException e) {
84.             if (!e.getMessage().contains("ENTRY_ALREADY_EXISTS")) {
85.                 System.out.println("Record already exist or an error occurred: " + e.
getMessage());
86.             }
87.         }
88.
89.         return true;
```

Issue ID	274544
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /LDAPManager.java
Line	117

Source Code

```
112.
113.         NamingEnumeration<SearchResult> results = ctx.search(base, filter, sc);
114.
115.         while (results.hasMore()) {
116.             SearchResult sr = (SearchResult) results.next();
117.             Attributes attrs = sr.getAttributes();
118.
119.             Attribute attr = attrs.get("uid");
120.             if (attr != null) {
121.                 // logger.debug("record found " + attr.get());
122.                 // System.out.println("record found " + attr.get());
```

Issue ID	274547
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	165

Source Code

```
160.
161.     public static String getCookie(HttpServletRequest request, String paramName) {
162.         Cookie[] values = request.getCookies();
163.         String param = "none";
164.         if (paramName != null) {
165.             for (int i = 0; i < values.length; i++) {
166.                 if (values[i].getName().equals(paramName)) {
167.                     param = values[i].getValue();
168.                     break; // break out of for loop when param found
169.                 }
170.             }
```

Issue ID	274560
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00001.java
Line	45

Source Code

```
40.         userCookie.setPath(request.getRequestURI());
41.         userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).
getHost());
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/pathtraver-00/BenchmarkTest00001.
html");
45.         rd.include(request, response);
46.     }
47.
```

```
48.    @Override
49.    public void doPost(HttpServletRequest request, HttpServletResponse response)
50.            throws ServletException, IOException {
```

Issue ID	274568
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00002.java
Line	45

Source Code

```
40.        userCookie.setPath(request.getRequestURI());
41.        userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.        response.addCookie(userCookie);
43.        javax.servlet.RequestDispatcher rd =
44.                request.getRequestDispatcher("/pathtraver-00/BenchmarkTest00002.html");
45.        rd.include(request, response);
46.    }
47.
48.    @Override
49.    public void doPost(HttpServletRequest request, HttpServletResponse response)
50.            throws ServletException, IOException {
```

Issue ID	274573
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00003.java
Line	45

Source Code

```
40.        userCookie.setPath(request.getRequestURI());
41.        userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
```

```
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/hash-00/BenchmarkTest00003.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274583
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00009.java
Line	46

Source Code

```
41.         // some code
42.         response.setContentType("text/html;charset=UTF-8");
43.
44.         String param = "";
45.         java.util.Enumeration<String> names = request.getHeaderNames();
46.         while (names.hasMoreElements()) {
47.             String name = (String) names.nextElement();
48.
49.             if (org.owasp.benchmark.helpers.Utils.commonHeaders.contains(name)) {
50.                 continue; // If standard header, move on to next one
51.             }
```

Issue ID	274582
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00009.java
Line	47

Source Code

```
42.         response.setContentType("text/html;charset=UTF-8");
43.
44.         String param = "";
45.         java.util.Enumeration<String> names = request.getHeaderNames();
46.         while (names.hasMoreElements()) {
47.             String name = (String) names.nextElement();
48.
49.             if (org.owasp.benchmark.helpers.Utils.commonHeaders.contains(name)) {
50.                 continue; // If standard header, move on to next one
51.             }
52.
```

Issue ID	274589
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00012.java
Line	72

Source Code

```
67.             javax.naming.NamingEnumeration<javax.naming.directory.SearchResult> results =
68.                     idc.search(base, filter, filters, sc);
69.             while (results.hasMore()) {
70.                 javax.naming.directory.SearchResult sr =
71.                         (javax.naming.directory.SearchResult) results.next();
72.                 javax.naming.directory.Attributes attrs = sr.getAttributes();
73.
74.                 javax.naming.directory.Attribute attr = attrs.get("uid");
75.                 javax.naming.directory.Attribute attr2 = attrs.get("street");
76.                 if (attr != null) {
77.                     response.getWriter()
```

Issue ID	274591
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00012.java

Line 84

Source Code

```
79.                                 "LDAP query results:<br>"
80.                                         + "Record found with name "
81.                                         + attr.get()
82.                                         + "<br>"
83.                                         + "Address: "
84.                                         + attr2.get()
85.                                         + "<br>");
86.                     // System.out.println("record found " + attr.get());
87.                     found = true;
88.                 }
89.             }
```

Issue ID 274598

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00021.java

Line 63

Source Code

```
58.             javax.naming.NamingEnumeration<javax.naming.directory.SearchResult>
results =
59.                     ctx.search(base, filter, filters, sc);
60.             while (results.hasMore()) {
61.                 javax.naming.directory.SearchResult sr =
62.                         (javax.naming.directory.SearchResult) results.next();
63.                 javax.naming.directory.Attributes attrs = sr.getAttributes();
64.
65.                 javax.naming.directory.Attribute attr = attrs.get("uid");
66.                 javax.naming.directory.Attribute attr2 = attrs.get("street");
67.                 if (attr != null) {
68.                     response.getWriter()
```

Issue ID	274599
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00021.java
Line	75

Source Code

```
70.                             "LDAP query results:<br>"
71.                                 + "Record found with name "
72.                                 + attr.get()
73.                                 + "<br>"
74.                                 + "Address: "
75.                                 + attr2.get()
76.                                 + "<br>");
77.                 // System.out.println("record found " + attr.get());
78.                 found = true;
79.             }
80.         }
```

Issue ID	274622
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00044.java
Line	64

Source Code

```
59.         javax.naming.NamingEnumeration<javax.naming.directory.SearchResult>
results =
60.                 ctx.search(base, filter, sc);
61.         while (results.hasMore()) {
62.             javax.naming.directory.SearchResult sr =
63.                     (javax.naming.directory.SearchResult) results.next();
64.             javax.naming.directory.Attributes attrs = sr.getAttributes();
65.
66.             javax.naming.directory.Attribute attr = attrs.get("uid");
```

```
67.         javax.naming.directory.Attribute attr2 = attrs.get("street");
68.         if (attr != null) {
69.           response.getWriter()
```

Issue ID	274620
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00044.java
Line	76

Source Code

```
71.                     "LDAP query results:<br>"
72.                         + "Record found with name "
73.                         + attr.get()
74.                         + "<br>"
75.                         + "Address: "
76.                         + attr2.get()
77.                         + "<br>");
78.           // System.out.println("record found " + attr.get());
79.           found = true;
80.         }
81.       }
```

Issue ID	274625
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00061.java
Line	45

Source Code

```
40.     userCookie.setPath(request.getRequestURI());
41.     userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.     response.addCookie(userCookie);
43.     javax.servlet.RequestDispatcher rd =
44.             request.getRequestDispatcher("/pathtraver-00/BenchmarkTest00061.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274630
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00062.java
Line	45

Source Code

```
40.         userCookie.setPath(request.getRequestURI());
41.         userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.             request.getRequestDispatcher("/pathtraver-00/BenchmarkTest00062.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274638
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00064.java
Line	45

Source Code

```
40.         userCookie.setPath(request.getRequestURI());
41.         userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/pathtraver-00/BenchmarkTest00064.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274640
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00065.java
Line	45

Source Code

```
40.         userCookie.setPath(request.getRequestURI());
41.         userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/pathtraver-00/BenchmarkTest00065.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274647
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00067.java
Line	45

Source Code

```
40.         userCookie.setPath(request.getRequestURI());
41.         userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/weakrand-00/BenchmarkTest00067.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274657
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00070.java
Line	45

Source Code

```
40.         userCookie.setPath(request.getRequestURI());
41.         userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/hash-00/BenchmarkTest00070.html");
45.         rd.include(request, response);
46.     }
47.
```

```
48.    @Override
49.    public void doPost(HttpServletRequest request, HttpServletResponse response)
50.            throws ServletException, IOException {
```

Issue ID	274662
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00076.java
Line	45

Source Code

```
40.        userCookie.setPath(request.getRequestURI());
41.        userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).
getHost());
42.        response.addCookie(userCookie);
43.        javax.servlet.RequestDispatcher rd =
44.                request.getRequestDispatcher("/hash-00/BenchmarkTest00076.html");
45.        rd.include(request, response);
46.    }
47.
48.    @Override
49.    public void doPost(HttpServletRequest request, HttpServletResponse response)
50.            throws ServletException, IOException {
```

Issue ID	274667
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00077.java
Line	45

Source Code

```
40.        userCookie.setPath(request.getRequestURI());
41.        userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).
getHost());
42.        response.addCookie(userCookie);
```

```
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/cmdi-00/BenchmarkTest00077.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274670
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00084.java
Line	45

Source Code

```
40.         userCookie.setPath(request.getRequestURI());
41.         userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.         response.addCookie(userCookie);
43.         javax.servlet.RequestDispatcher rd =
44.                 request.getRequestDispatcher("/weakrand-00/BenchmarkTest00084.html");
45.         rd.include(request, response);
46.     }
47.
48.     @Override
49.     public void doPost(HttpServletRequest request, HttpServletResponse response)
50.             throws ServletException, IOException {
```

Issue ID	274680
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00086.java
Line	45

Source Code

```
40.        userCookie.setPath(request.getRequestURI());
41.        userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.        response.addCookie(userCookie);
43.        javax.servlet.RequestDispatcher rd =
44.                request.getRequestDispatcher("/weakrand-00/BenchmarkTest00086.html");
45.        rd.include(request, response);
46.    }
47.
48.    @Override
49.    public void doPost(HttpServletRequest request, HttpServletResponse response)
50.            throws ServletException, IOException {
```

Issue ID	274690
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00087.java
Line	45

Source Code

```
40.        userCookie.setPath(request.getRequestURI());
41.        userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
42.        response.addCookie(userCookie);
43.        javax.servlet.RequestDispatcher rd =
44.                request.getRequestDispatcher("/securecookie-00/BenchmarkTest00087.html");
45.        rd.include(request, response);
46.    }
47.
48.    @Override
49.    public void doPost(HttpServletRequest request, HttpServletResponse response)
50.            throws ServletException, IOException {
```

Issue ID	274696
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00092.java
Line	45

Source Code

```
40.        userCookie.setPath(request.getRequestURI());
41.        userCookie.setDomain(new java.net.URL(request.getRequestURL().toString()).
getHost());
42.        response.addCookie(userCookie);
43.        javax.servlet.RequestDispatcher rd =
44.                request.getRequestDispatcher("/cmdi-00/BenchmarkTest00092.html");
45.        rd.include(request, response);
46.    }
47.
48.    @Override
49.    public void doPost(HttpServletRequest request, HttpServletResponse response)
50.            throws ServletException, IOException {
```

● [Rule Name] Improper exception handling (High, Java)

The Improper exception handling checker finds instances of generalizing a wide range of exceptions in handling them.

For example, if an exception of root class such as Throwable or Exception is handled, then it can be considered as a generalized exception handling.

An extremely wide generalization of exceptions may prevent some specific exceptions from being properly handled; more seriously, this may let even logically implausible cases simply covered by the super generalized handler, hence hindering them from being detected and fixed. As a result, design and implementation defects may remain undiscovered in the development and review phases before the program is deployed into production, possibly leading to serious issues.

It is recommended to categorize possible exceptions explicitly and handle them separately. If general exception handling is needed, have it performed in the last place after all specific exceptions are handled. Then, if the flow of execution ever reaches the general exception handling block, it suggests that the caught exception was not considered during development. This helps fix defects implied by unexpected exceptions.

- CWE 660 4.14
  - 397 - Declaration of Throws for Generic Exception
- CWE 660 4.7
  - Declaration of Throws for Generic Exception - (397)
- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-755
- 소프트웨어 보안약점 진단가이드 2021
  - 부적절한 예외처리

Dangerous Example

```
1. public void readFromFile(String fileName){
2.   try{
3.     File myFile = new File(fileName);
4.     FileReader fr = new FileReader(myFile);
5.   } catch(Exception ex){ }
6. }
```

Line 5: A generalized exception Exception is handled instead of detailed input / output exceptions of using file related APIs.

Safe Example

```
1. public void readFromFile(String fileName) throws FileNotFoundException,
IOException, MyException {
2.   try {
3.     //Null check for fileName
4.    if(fileName == NULL) throw new MyException("error");
5.    File myFile = new File(fileName);
6.    FileReader fr = new FileReader(myFile);
7.  } catch(FileNotFoundException fe){
8.    ...
9.  } catch(IOException ie){
10.    ...
11.  }
12. }
```

Line 7: Explicitly handle the detailed input / output exceptions.

Issue ID	274466
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java
Line	112

Source Code

```
107.                       + "END;");
108.            conn.commit();
109.            initData();
110.
111.            System.out.println("DataBase tables/procedures created.");
112.        } catch (Exception e1) {
113.            System.out.println(
114.                    "Problem with database table/procedure creations: " + e1.
getMessage());
115.        }
116.    }
117.
```

Issue ID	274464
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java

Line	151

Source Code

```
146.        executeSQLCommand("INSERT INTO SCORE (nick, score) VALUES('foo', 40)");
147.
148.        executeSQLCommand(
149.            "INSERT INTO EMPLOYEE (first_name, last_name, salary) VALUES ('foo', 'bar', 34567)");
150.        conn.commit();
151.      } catch (Exception e1) {
152.        System.out.println("Problem with database init/reset: " + e1.getMessage());
153.      }
154.    }
155.
156.    public static java.sql.Connection getSqlConnection() {
```

Issue ID	274471
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/LDAPManager.java
Line	128

Source Code

```
123.              }
124.          }
125.          ctx.close();
126.
127.          return true;
128.      } catch (Exception e) {
129.          System.out.println("LDAP error search: ");
130.          e.printStackTrace();
131.          return false;
132.      }
133.    }
```

Issue ID	274475
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/LDAPServer.java
Line	102

Source Code

```
97.           File workDir = new File(dir + "/../ldap");
98.           workDir.mkdirs();
99.           System.setProperty("workingDiretory", workDir.getPath());
100.
101.            init();
102.        } catch (Exception e) {
103.            System.out.println("Error initializing LDAP Server: " + e.getMessage());
104.            e.printStackTrace();
105.        }
106.
107.        LDAPManager emd = new LDAPManager();
```

Issue ID	274485
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	326

Source Code

```
321.                BufferedReader br = new BufferedReader(fr); ) {
322.            String line;
323.            while ((line = br.readLine()) != null) {
324.                sourceLines.add(line);
325.            }
326.        } catch (Exception e) {
327.            try {
328.                System.out.println("Problem reading contents of file: " + file.
getCanonicalFile());
```

```
329.          } catch (IOException e2) {
330.              System.out.println("Problem reading file to get lines from.");
331.              e2.printStackTrace();
```

Issue ID	274490
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	84

Source Code

```
79.                             + org.owasp
80.                                     .esapi
81.                                     .ESAPI
82.                                     .encoder()
83.                                     .encodeForHTML(new String(b, 0, size)));
84.         } catch (Exception e) {
85.             System.out.println("Couldn't open FileInputStream on file: '" + fileName +
"'");
86.             response.getWriter()
87.                     .println(
88.                             "Problem getting FileInputStream: "
89.                                     + org.owasp
```

Issue ID	274491
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	99

Source Code

```
94.         } finally {
95.             if (fis != null) {
96.                 try {
97.                     fis.close();
98.                 fis = null;
99.             } catch (Exception e) {
100.                // we tried...
101.            }
102.        }
103.    }
104. }
```

Issue ID	274493
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00002.java
Line	78

Source Code

```
73.             response.getWriter()
74.                     .println(
75.                             "Now ready to write to file: "
76.                                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML
(fileName));
77.
78.         } catch (Exception e) {
79.             System.out.println("Couldn't open FileOutputStream on file: '" + fileName
+ "'");
80.             //                      System.out.println("File exception caught and swallowed:
" + e.getMessage());
81.         } finally {
82.             if (fos != null) {
83.                 try {
```

Issue ID	274494
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00002.java
Line	86

Source Code

```
81.         } finally {
82.             if (fos != null) {
83.                 try {
84.                     fos.close();
85.                     fos = null;
86.                 } catch (Exception e) {
87.                     // we tried...
88.                 }
89.             }
90.         }
91.     }
```

Issue ID	274499
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00012.java
Line	101

Source Code

```
96.         } catch (javax.naming.NamingException e) {
97.             throw new ServletException(e);
98.         } finally {
99.             try {
100.                ads.closeDirContext();
101.            } catch (Exception e) {
102.                throw new ServletException(e);
103.            }
104.        }
105.    }
106. }
```

Issue ID	274510
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00021.java

Line	92

Source Code

```
87.         } catch (javax.naming.NamingException e) {
88.             throw new ServletException(e);
89.         } finally {
90.             try {
91.                 ads.closeDirContext();
92.             } catch (Exception e) {
93.                 throw new ServletException(e);
94.             }
95.         }
96.     }
97. }
```

Issue ID	274512
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00028.java
Line	63

Source Code

```
58.             response.getWriter()
59.                     .println(
60.                             "Now ready to write to file: "
61.                                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML
(fileName));
62.
63.         } catch (Exception e) {
64.             System.out.println("Couldn't open FileOutputStream on file: '" + fileName
+ "'");
65.             //                    System.out.println("File exception caught and swallowed:
" + e.getMessage());
66.         } finally {
67.             if (fos != null) {
68.                 try {
```

Issue ID	274513
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00028.java
Line	71

Source Code

```
66.        } finally {
67.            if (fos != null) {
68.                try {
69.                    fos.close();
70.                    fos = null;
71.                } catch (Exception e) {
72.                    // we tried...
73.                }
74.            }
75.        }
76.    }
```

Issue ID	274515
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00044.java
Line	93

Source Code

```
88.        } catch (javax.naming.NamingException e) {
89.            throw new ServletException(e);
90.        } finally {
91.            try {
92.                ads.closeDirContext();
93.            } catch (Exception e) {
94.                throw new ServletException(e);
95.            }
96.        }
```

```
97.     }
98. }
```

Issue ID	274525
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00062.java
Line	90

Source Code

```
85.                                     + org.owasp
86.                                             .esapi
87.                                             .ESAPI
88.                                             .encoder()
89.                                             .encodeForHTML(new String(b, 0, size)));
90.         } catch (Exception e) {
91.             System.out.println("Couldn't open FileInputStream on file: '" + fileName + "'");
92.             response.getWriter()
93.                     .println(
94.                             "Problem getting FileInputStream: "
95.                                     + org.owasp
```

Issue ID	274526
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00062.java
Line	105

Source Code

```
100.        } finally {
101.            if (fis != null) {
102.                try {
103.                    fis.close();
104.                    fis = null;
105.             } catch (Exception e) {
106.                 // we tried...
107.             }
108.         }
109.     }
110. }
```

Issue ID	274528
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00064.java
Line	84

Source Code

```
79.         response.getWriter()
80.                 .println(
81.                         "Now ready to write to file: "
82.                                 + org.owasp.esapi.ESAPI.encoder().encodeForHTML
(fileName));
83.
84.     } catch (Exception e) {
85.         System.out.println("Couldn't open FileOutputStream on file: '" + fileName
+ "'");
86.         //                  System.out.println("File exception caught and swallowed:
" + e.getMessage());
87.     } finally {
88.         if (fos != null) {
89.             try {
```

Issue ID	274529
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00064.java
Line	92

Source Code

```
87.         } finally {
88.             if (fos != null) {
89.                 try {
90.                     fos.close();
91.                     fos = null;
92.                 } catch (Exception e) {
93.                     // we tried...
94.                 }
95.             }
96.         }
97.     }
```

Issue ID	274531
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00065.java
Line	90

Source Code

```
85.                                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML(fileName)
86.                                     + "' is:₩n₩n");
87.             response.getWriter()
88.                     .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(new String(b, 0, size)));
89.             is.close();
90.         } catch (Exception e) {
91.             System.out.println("Couldn't open InputStream on file: '" + fileName + "'");
92.             response.getWriter()
93.                     .println(
94.                             "Problem getting InputStream: "
95.                                     + org.owasp
```

Issue ID	274532
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode

File /BenchmarkTest00065.java

Line 105

Source Code

```
100.        } finally {
101.            if (is != null) {
102.                try {
103.                    is.close();
104.                    is = null;
105.                } catch (Exception e) {
106.                    // we tried...
107.                }
108.            }
109.        }
110.    }
```

● [Rule Name] Empty catch block (Medium, Java)

The Empty catch block checker finds exception-handling blocks that have no relevant content.

It detects empty catch blocks containing no actual executable code.

If a caught exception is not handled, it is hard to find out the causes of accompanying program errors.

Add an exception-handling code in the empty block. If you don''t want any measure for an exception, you may simply leave an error message.

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-390

- 소프트웨어 보안약점 진단가이드 2021
  - 오류 상황 대응 부재

Dangerous Example

```
1. private Connection conn;
2. public Connection DBConnect(String url, String id, String password) {
3.   try {
4.     String CONNECT_STRING = url + ":" + id + ":" + password;
5.     InitialContext ctx = new InitialContext();
6.     DataSource datasource = (DataSource) ctx.lookup(CONNECT_STRING);
7.     conn = datasource.getConnection();
8.   } catch (SQLException e) {
9.     // Catch block is empty
10.   } catch (NamingException e) {
11.     // Catch block is empty
12.   }
13.   return conn;
14. }
```

Line 8: An exception is caught in a catch block, but no action is taken.
Line 10: An exception is caught in a catch block, but no action is taken.

Safe Example

```
private Connection conn;
public Connection DBConnect(String url, String id, String password) {
  try {
    String CONNECT_STRING = url + ":" + id + ":" + password;
    InitialContext ctx = new InitialContext();
    DataSource datasource = (DataSource) ctx.lookup(CONNECT_STRING);
    conn = datasource.getConnection();
  } catch (SQLException e) {
    // Proper Exception handling
    if ( conn != null ) {
      try {
        conn.close();
      } catch (SQLException e1) {
        conn = null;
      }
    }
  } catch (NamingException e) {
    // Proper Exception handling.
    if ( conn != null ) {
      try {
```

```
21.       conn.close();
22.     } catch (SQLException e1) {
23.       conn = null;
24.     }
25.   }
26. }
27. return conn;
28. }
```

Line 10: Take an action for the caught exception.
Line 17: Take an action for the caught exception.

Issue ID	274489
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	99

Source Code

```
94.         } finally {
95.             if (fis != null) {
96.                 try {
97.                     fis.close();
98.                     fis = null;
99.                 } catch (Exception e) {
100.                    // we tried...
101.                }
102.            }
103.        }
104.    }
```

Issue ID	274492
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00002.java
Line	86

Source Code

```
81.         } finally {
82.             if (fos != null) {
83.                 try {
84.                     fos.close();
85.                     fos = null;
86.                 } catch (Exception e) {
87.                     // we tried...
88.                 }
89.             }
90.         }
91.     }
```

Issue ID	274511
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00028.java
Line	71

Source Code

```
66.         } finally {
67.             if (fos != null) {
68.                 try {
69.                     fos.close();
70.                     fos = null;
71.                 } catch (Exception e) {
72.                     // we tried...
73.                 }
74.             }
75.         }
76.     }
```

Issue ID	274524
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode

File	/BenchmarkTest00062.java
Line	105

Source Code

```
100.        } finally {
101.            if (fis != null) {
102.                try {
103.                    fis.close();
104.                    fis = null;
105.                } catch (Exception e) {
106.                    // we tried...
107.                }
108.            }
109.        }
110.    }
```

Issue ID	274527
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00064.java
Line	92

Source Code

```
87.        } finally {
88.            if (fos != null) {
89.                try {
90.                    fos.close();
91.                    fos = null;
92.                } catch (Exception e) {
93.                    // we tried...
94.                }
95.            }
96.        }
97.    }
```

Issue ID	274530
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00065.java
Line	105

Source Code

```
100.         } finally {
101.             if (is != null) {
102.                 try {
103.                     is.close();
104.                     is = null;
105.                 } catch (Exception e) {
106.                     // we tried...
107.                 }
108.             }
109.         }
110.     }
```

● [Rule Name] Violation of Trust Boundary (Critical, Java)

The Violation of Trust Boundary checker finds instances of mixing up trusted and untrusted data.

Sending an untrusted external input as a session value or attribute could be an example.

If trusted and untrusted data is mixed, the programmer may trust untrusted data by mistake. The boundary of trusted data can be thought of as a separating line: On one side of the line lies untrusted data. Data on the other side can be trusted. Data validation is a process of moving data over this boundary line, i.e., from the untrusted side to the trusted side. In case the two sides are blended somehow, the trust boundary is violated. This often happens in a data structure that contains trusted and untrusted data together.

Trusted data must be managed in a separate storage. Any external inputs should be validated before deposited in this storage.

Dangerous Example

```
javax.servlet.http.Cookie[] theCookies = request.getCookies();
String param = "";
```

```
3. if (theCookies != null) {
4.   for (javax.servlet.http.Cookie theCookie : theCookies) {
5.     if (theCookie.getName().equals("danger")) {
6.       param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
7.       break;
8.     }
9.   }
10. }
11.
12. request.getSession().setAttribute( param, "danger param"); //Bad
```

Line 12: An untrusted value sent by cookies in request is directly stored to session.

Safe Example

```
1. javax.servlet.http.Cookie[] theCookies = request.getCookies();
2. String param = "";
3. if (theCookies != null) {
4.   for (javax.servlet.http.Cookie theCookie : theCookies) {
5.     if (theCookie.getName().equals("danger")) {
6.       param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
7.       break;
8.     }
9.   }
10. }
11.
12. if(isSafe(param)){
13.   request.getSession().setAttribute( param, "danger param"); //Good
14. else {
15.   //
16. }
```

Line 12: Use the external input after checking its desired form.

Issue ID	274606
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00023.java

Line 82

Source Code

```
77.         rememberMe.setSecure(true);
78.         rememberMe.setHttpOnly(true);
79.         rememberMe.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
80.         rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
81.         // e.g., /benchmark/sql-01/BenchmarkTest01001
82.         request.getSession().setAttribute(cookieName, rememberMeKey);
83.         response.addCookie(rememberMe);
84.         response.getWriter()
85.             .println(
86.                 user
87.                     + " has been remembered with cookie: "
```

Issue ID	274615
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00042.java
Line	85

Source Code

```
80.                 new javax.servlet.http.Cookie(cookieName, rememberMeKey);
81.         rememberMe.setSecure(true);
82.         rememberMe.setHttpOnly(true);
83.         rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
84.         // e.g., /benchmark/sql-01/BenchmarkTest01001
85.         request.getSession().setAttribute(cookieName, rememberMeKey);
86.         response.addCookie(rememberMe);
87.         response.getWriter()
88.             .println(
89.                 user
90.                     + " has been remembered with cookie: "
```

Issue ID	274650
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00067.java
Line	124

Source Code

```
119.          rememberMe.setSecure(true);
120.          rememberMe.setHttpOnly(true);
121.          rememberMe.setDomain(new java.net.URL(request.getRequestURL().
toString()).getHost());
122.          rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST
this servlet
123.          // e.g., /benchmark/sql-01/BenchmarkTest01001
124.          request.getSession().setAttribute(cookieName, rememberMeKey);
125.          response.addCookie(rememberMe);
126.          response.getWriter()
127.                  .println(
128.                          user
129.                                  + " has been remembered with cookie: "
```

Issue ID	274671
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00084.java
Line	103

Source Code

```
98.                 new javax.servlet.http.Cookie(cookieName, rememberMeKey);
99.           rememberMe.setSecure(true);
100.          rememberMe.setHttpOnly(true);
101.          rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST
this servlet
102.          // e.g., /benchmark/sql-01/BenchmarkTest01001
103.          request.getSession().setAttribute(cookieName, rememberMeKey);
```

```
104.          response.addCookie(rememberMe);
105.          response.getWriter()
106.                  .println(
107.                          user
108.                                  + " has been remembered with cookie: "
```

Issue ID	274686
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00086.java
Line	107

Source Code

```
102.                  new javax.servlet.http.Cookie(cookieName, rememberMeKey);
103.          rememberMe.setSecure(true);
104.          rememberMe.setHttpOnly(true);
105.          rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
106.          // e.g., /benchmark/sql-01/BenchmarkTest01001
107.          request.getSession().setAttribute(cookieName, rememberMeKey);
108.          response.addCookie(rememberMe);
109.          response.getWriter()
110.                  .println(
111.                          user
112.                                  + " has been remembered with cookie: "
```

● [Rule Name] Generating predictable random value (High, Java)

The Generating predictable random value checker finds instances of using a predictable random value.

If a predictable random number is used when an unpredictable number is demanded, an attacker can predict subsequently generated numbers and use them to attack the system. Languages provide their specific built-in statistical PRNG (pseudorandom number generator) functionality. But numbers generated by such a generator are often easily predictable, and security can be threatened simply by an effective seed setting.

You need to use cryptographically secure approaches that generate unpredictable random values. The SecureRandom class enables creating secure random values.

- 무기체계 소프트웨어 보안약점 점검 목록
    - CWE-330

- 소프트웨어 보안약점 진단가이드 2021
    - 적절하지 않은 난수 값 사용

Dangerous Example

```
1. public double roledice() {
2.   return Math.random();
3. }
```

Line 2: The random () method of the java.lang.Math class is dangerous because the seed cannot be reset. Moreover, java.util.Random class can reset the seed.

Safe Example

```
1. import java.security.SecureRandom;
2. import java.util.Random;
3. import java.util.Date;
4. public int roledice() {
5.   Random numGen = SecureRandom.getInstance("SHA1PRNG");
6.   return (numGen.nextInt(6)) + 1;
7. }
```

Line 6: Create the unpredictable random number by setting a seed by SecureRandom class than using java.util.Random.

Issue ID	274605
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00023.java
Line	47

Source Code

```
42.        response.setContentType("text/html;charset=UTF-8");
43.
44.        String param = request.getParameter("BenchmarkTest00023");
45.        if (param == null) param = "";
46.
47.        float rand = new java.util.Random().nextFloat();
48.        String rememberMeKey = Float.toString(rand).substring(2); // Trim off the 0. at the front.
49.
50.        String user = "Floyd";
51.        String fullClassName = this.getClass().getName();
52.        String testCaseNumber =
```

Issue ID	274672
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00084.java
Line	69

Source Code

```
64.
65.        org.owasp.benchmark.helpers.ThingInterface thing =
66.                org.owasp.benchmark.helpers.ThingFactory.createThing();
67.        String bar = thing.doSomething(param);
68.
69.        int r = new java.util.Random().nextInt();
70.        String rememberMeKey = Integer.toString(r);
71.
72.        String user = "Ingrid";
73.        String fullClassName = this.getClass().getName();
74.        String testCaseNumber =
```

Issue ID	274687

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00086.java

Line 72

Source Code

```
67.    // Simple if statement that assigns constant to bar on true condition
68.    int num = 86;
69.    if ((7 * 42) - num > 200) bar = "This_should_always_happen";
70.    else bar = param;
71.
72.    long l = new java.util.Random().nextLong();
73.    String rememberMeKey = Long.toString(l);
74.
75.    String user = "Logan";
76.    String fullClassName = this.getClass().getName();
77.    String testCaseNumber =
```

● [Rule Name] Weak cryptographic algorithm (High, Java)

The Weak Cryptographic Algorithm checker finds instances of using a cryptographic algorithm known to be insecure.

Non-standard cryptographic algorithms can be more easily cryptanalyzed and cracked by attackers. Some old algorithms have been weak over time as computing power grows-It used to be expected to take an extremely long time to crack them in the past, but now they can be broken in a couple of days or hours. Using an old or non-standard cryptographic algorithm may allow attackers to cryptanalyze and disable it.

Developing your own cryptographic algorithm is dangerous; instead, use standard algorithms that have been proven in academia and industry. Use secure algorithms such as 3DES, AES, and SEED instead of those considered weak such as DES and RC5. And encryption keys must be long enough in accordance with the appropriate length suggested by each standard, secure algorithm.

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-327

- 소프트웨어 보안약점 진단가이드 2021
  - 취약한 암호화 알고리즘 사용

Dangerous Example

```
1. import java.security.*;
2. import javax.crypto.Cipher;
3. import javax.crypto.NoSuchPaddingException;
4. public class CryptoUtils {
5.   public byte[] encrypt(byte[] msg, Key k) {
6.     byte[] rslt = null;
7.     try {
8.       Cipher c = Cipher.getInstance("DES");
9.       c.init(Cipher.ENCRYPT_MODE, k);
10.       rslt = c.update(msg);
11.     } catch (InvalidKeyException e) {
12.       System.err.println("Exception occured!");
13.     } catch (NoSuchAlgorithmException e) {
14.       System.err.println("Exception occured!");
15.     } catch (NoSuchPaddingException e) {
16.       System.err.println("Exception occured!");
17.     }
18.     return rslt;
19.   }
20. }
```

Line 8: An encryption is performed with a weak DES algorithm.

Safe Example

```
1. import java.security.*;
2. import javax.crypto.Cipher;
3. import javax.crypto.NoSuchPaddingException;
4. public class CryptoUtils {
5.   public byte[] encrypt(byte[] msg, Key k) {
6.     byte[] rslt = null;
7.     try {
8.       Cipher c = Cipher.getInstance("AES/CBC/PKCS5Padding");
9.       c.init(Cipher.ENCRYPT_MODE, k);
10.       rslt = c.update(msg);
11.     } catch (InvalidKeyException e) {
12.       System.err.println("Exception occured!");
13.     } catch (NoSuchAlgorithmException e) {
14.       System.err.println("Exception occured!");
15.     } catch (NoSuchPaddingException e) {
16.       System.err.println("Exception occured!");
17.     }
18.     return rslt;
19.   }
20. }
```

Line 8: Use AES algorithms known to be secure.

Issue ID	274496
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00005.java
Line	63

Source Code

```
58.         //              };
59.         java.security.SecureRandom random = new java.security.SecureRandom();
60.         byte[] iv = random.generateSeed(8); // DES requires 8 byte keys
61.
62.         try {
63.             javax.crypto.Cipher c = javax.crypto.Cipher.getInstance("DES/CBC
/PKCS5Padding");
64.
65.             // Prepare the cipher to encrypt
66.             javax.crypto.SecretKey key = javax.crypto.KeyGenerator.getInstance("DES").
generateKey();
67.             java.security.spec.AlgorithmParameterSpec paramSpec =
68.                     new javax.crypto.spec.IvParameterSpec(iv);
```

Issue ID	274497

File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00005.java
Line	66

Source Code

```
61.
62.        try {
63.            javax.crypto.Cipher c = javax.crypto.Cipher.getInstance("DES/CBC
/PKCS5Padding");
64.
65.            // Prepare the cipher to encrypt
66.            javax.crypto.SecretKey key = javax.crypto.KeyGenerator.getInstance("DES").
generateKey();
67.            java.security.spec.AlgorithmParameterSpec paramSpec =
68.                    new javax.crypto.spec.IvParameterSpec(iv);
69.            c.init(javax.crypto.Cipher.ENCRYPT_MODE, key, paramSpec);
70.
71.            // encrypt and store the results
```

Issue ID	274508
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00020.java
Line	59

Source Code

```
54.        java.security.SecureRandom random = new java.security.SecureRandom();
55.        byte[] iv = random.generateSeed(8); // DES requires 8 byte keys
56.
57.        try {
58.            javax.crypto.Cipher c =
59.                    javax.crypto.Cipher.getInstance("DES/CBC/PKCS5Padding", "SunJCE");
60.            // Prepare the cipher to encrypt
61.            javax.crypto.SecretKey key = javax.crypto.KeyGenerator.getInstance("DES").
generateKey();
62.            java.security.spec.AlgorithmParameterSpec paramSpec =
63.                 new javax.crypto.spec.IvParameterSpec(iv);
64.           c.init(javax.crypto.Cipher.ENCRYPT_MODE, key, paramSpec);
```

Issue ID	274509
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00020.java
Line	61

Source Code

```
56.
57.        try {
58.           javax.crypto.Cipher c =
59.                 javax.crypto.Cipher.getInstance("DES/CBC/PKCS5Padding", "SunJCE");
60.           // Prepare the cipher to encrypt
61.           javax.crypto.SecretKey key = javax.crypto.KeyGenerator.getInstance("DES").
generateKey();
62.           java.security.spec.AlgorithmParameterSpec paramSpec =
63.                 new javax.crypto.spec.IvParameterSpec(iv);
64.           c.init(javax.crypto.Cipher.ENCRYPT_MODE, key, paramSpec);
65.
66.           // encrypt and store the results
```

Issue ID	274522
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00050.java
Line	60

Source Code

```
55.        java.security.SecureRandom random = new java.security.SecureRandom();
56.        byte[] iv = random.generateSeed(8); // DES requires 8 byte keys
57.
58.        try {
59.           javax.crypto.Cipher c =
60.                 javax.crypto.Cipher.getInstance("DES/CBC/PKCS5Padding", "SunJCE");
61.             // Prepare the cipher to encrypt
62.             javax.crypto.SecretKey key = javax.crypto.KeyGenerator.getInstance("DES").
generateKey();
63.             java.security.spec.AlgorithmParameterSpec paramSpec =
64.                     new javax.crypto.spec.IvParameterSpec(iv);
65.             c.init(javax.crypto.Cipher.ENCRYPT_MODE, key, paramSpec);
```

Issue ID	274523
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00050.java
Line	62

Source Code

```
57.
58.         try {
59.             javax.crypto.Cipher c =
60.                     javax.crypto.Cipher.getInstance("DES/CBC/PKCS5Padding", "SunJCE");
61.             // Prepare the cipher to encrypt
62.             javax.crypto.SecretKey key = javax.crypto.KeyGenerator.getInstance("DES").
generateKey();
63.             java.security.spec.AlgorithmParameterSpec paramSpec =
64.                     new javax.crypto.spec.IvParameterSpec(iv);
65.             c.init(javax.crypto.Cipher.ENCRYPT_MODE, key, paramSpec);
66.
67.             // encrypt and store the results
```

Issue ID	274534
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00070.java
Line	73

Source Code

```
        int num = 106;

        bar = (7 * 42) - num > 200 ? "This should never happen" : param;

        try {
            java.security.MessageDigest md = java.security.MessageDigest.getInstance("SHA1", "SUN");
            byte[] input = {(byte) '?'};
            Object inputParam = bar;
            if (inputParam instanceof String) input = ((String) inputParam).getBytes();
            if (inputParam instanceof java.io.InputStream) {
                byte[] strInput = new byte[1000];
```

● [Rule Name] SQL Injection (High, Java)

The SQL Injection checker finds SQL queries that include an unvalidated external input.

Failure to validate input data received from a Web application may allow attackers to inject SQL queries through input forms or the address bar, thereby accessing or manipulating information in the database.

Use a PreparedStatement object to send precompiled query statements (constants) to the database. PreparedStatement enables filtering database queries for special characters and reserved words.

- 무기체계 소프트웨어 보안약점 점검 목록
    - CWE-89

- 소프트웨어 보안약점 진단가이드 2021
    - SQL 삽입

Dangerous Example

```
String query = "SELECT account_balance FROM"
               + "user_data WHERE user_name = "
               + request.getParameter("customerName");
try {
```

```
5.   Statement statement = connection.createStatement( ... );
6.   ResultSet results = statement.executeQuery(query);
7. }
```

Line 3: An external input, request.getParameter("customerName") is included in a query without validation.
Line 6: A query is passed as an argument of statement.executeQuery(). Which enables attackers to execute commands on database.

Safe Example

```
1. String custname = request.getparameter("customerName"); // Verification required
2. // perform input validation to detect attacks
3. String query = "SELECT account_balance FROM user_data WHERE user_name = ?";
4. PreparedStatement pstmt = connection.prepareStatement(query);
5. pstmt.setString(1, custname);
6. ResultSet results = pstmt.executeQuery();
```

Line 5: This prevent the query from being changed by preparedStatement() even when attackers inject the SQL commands.

Issue ID	274593
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00018.java
Line	59

Source Code

```
54.        String sql = "INSERT INTO users (username, password) VALUES ('foo','" + param + "')";
55.
56.        try {
57.            java.sql.Statement statement =
58.                    org.owasp.benchmark.helpers.DatabaseHelper.getSqlStatement();
59.            int count = statement.executeUpdate(sql);
60.            org.owasp.benchmark.helpers.DatabaseHelper.outputUpdateComplete(sql, response);
61.        } catch (java.sql.SQLException e) {
```

```
62.            if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
63.                response.getWriter().println("Error processing request.");
64.                return;
```

Issue ID	274612
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00034.java
Line	56

Source Code

```
51.        String sql = "SELECT * from USERS where USERNAME='foo' and PASSWORD='" + param + "'";
52.
53.        try {
54.            java.sql.Statement statement =
55.                    org.owasp.benchmark.helpers.DatabaseHelper.getSqlStatement();
56.            statement.execute(sql, java.sql.Statement.RETURN_GENERATED_KEYS);
57.            org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, response);
58.        } catch (java.sql.SQLException e) {
59.            if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
60.                response.getWriter().println("Error processing request.");
61.                return;
```

● [Rule Name] Critical public variable without final modifier (High, Java)

The Critical public variable without final modifier checker finds instances in which a critical member field that should not be arbitrarily modified is declared public static without the final modifier.

Because the checker cannot determine which fields are considered critical by the programmer, it regards primitive-type fields as critical, which are usually used as global constants.

Fields that are not declared final can be arbitrarily modified from the outside.

When declaring primitive fields as public static, add the final modifier. In the case of critical fields that need to be mutable, do not declare them public and have them accessed via static methods.

- CWE 660 4.14
  - 493 - Critical Public Variable Without Final Modifier
  - 500 - Public Static Field Not Marked Final
- CWE 660 4.7
  - Critical Public Variable Without Final Modifier - (493)
  - Public Static Field Not Marked Final - (500)

Dangerous Example

```
1. public final class myClass extends AppleIt{
2.   // var field is not final
3.   // value of var can be modified from external source.
4.   public static int var = 20;
5.   public int getTotal(int n){
6.   return var * n;
7. }
```

Line 4: A final modifier is not specified for var variable that is declared as public static.

Safe Example

```
1. public final class myClass extends AppleIt{
2.   // declare variables using keyword final
3.   // if it should not be altered
4.   public static final int var = 20;
5.   public int getTotal(int n){
6.   return var * n;
7. }
```

Line 4: Declare a final modifier in important public variables.

Issue ID	274458
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java
Line	37

Source Code

```
32. import org.owasp.benchmark.service.pojo.XMLMessage;
33. import org.owasp.esapi.ESAPI;
34.
35. public class DatabaseHelper {
36.     private static Connection conn;
37.     public static org.springframework.jdbc.core.JdbcTemplate JDBCtemplate;
38.     public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtil =
39.             new org.owasp.benchmark.helpers.HibernateUtil(false);
40.     public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtilClassic =
41.             new org.owasp.benchmark.helpers.HibernateUtil(true);
42.     public static final boolean hideSQLErrors =
```

Issue ID	274459
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java
Line	38

Source Code

```
33. import org.owasp.esapi.ESAPI;
34.
35. public class DatabaseHelper {
36.     private static Connection conn;
37.     public static org.springframework.jdbc.core.JdbcTemplate JDBCtemplate;
38.     public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtil =
39.             new org.owasp.benchmark.helpers.HibernateUtil(false);
40.     public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtilClassic =
41.             new org.owasp.benchmark.helpers.HibernateUtil(true);
42.     public static final boolean hideSQLErrors =
43.         false; // If we want SQL Exceptions to be suppressed from being displayed
to the user of
```

Issue ID	274460
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java
Line	40

Source Code

```
35. public class DatabaseHelper {
36.     private static Connection conn;
37.     public static org.springframework.jdbc.core.JdbcTemplate JDBCtemplate;
38.     public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtil =
39.             new org.owasp.benchmark.helpers.HibernateUtil(false);
40.     public static org.owasp.benchmark.helpers.HibernateUtil hibernateUtilClassic =
41.             new org.owasp.benchmark.helpers.HibernateUtil(true);
42.     public static final boolean hideSQLErrors =
43.             false; // If we want SQL Exceptions to be suppressed from being displayed
to the user of
44.     // the web app.
45.
```

● [Rule Name] Command injection (High, Java)

The Command injection checker finds instances in which an internal system command is executed by an unvalidated external input.

If a user input that has not been properly validated constitutes the whole or part of an OS command to execute it in an unintended way, this may cause improper permission changes or harmful effects on the system''s processes or operations.

If a command needs to be generated or selected based on an external input, whitelist safe values for command generation and restrict values determined by external inputs to those selected from within the whitelist.

- OWASP 2017

  - A1-Injection

- OWASP 2021
  - A03 Injection

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-78

- 소프트웨어 보안약점 진단가이드 2021
  - 운영체제 명령어 삽입

Dangerous Example

```
1. public void foo() throws IOException{
2.   Properties props  = new Properties();
3.   String filename = "file_list";
4.   FileInputStream in = new FileInputStream(fileName);
5.   props.load(in);
6.   String version = props.getProperty("dir_type");
7.   // Unusual behavior if dir_type is an unintended string
8.   String cmd = new String("cmd.exe /K  ₩"rmanDB.bat ₩"");
9.   Runtime.getRuntime().exec(cmd + "c:₩₩prog_cmd₩₩" + version);
10. }
```

Line 9: An external input props.getProperty("dir_type") is used to run commands without validation.

Safe Example

```
1. public void foo() throws IOException{
2.   Properties props  = new Properties();
3.   String filename = "file_list";
4.   FileInputStream in = new FileInputStream(fileName);
5.   props.load(in);
6.   String version[] = {"1.0", "1.0.1", "1.11", "1.4"};
7.  int versionSelection = Integer.parseInt(props.getProperty("version"));
8.  String cmd = new String("cmd.exe /K  ₩"rmanDB.bat ₩"");
9.  String vs = "";
10.   if(versionSelection == 0)
11.     vs = version[0];
12.   else if(versionSelection == 1)
13.     vs = version[1];
14.   else if(versionSelection == 2)
15.     vs = version[2];
16.   else if(versionSelection == 3)
17.     vs = version[3];
18.   else
19.     vs = vsersion[3];
20.   Runtime.getRuntime().exec(cmd + "c:₩₩prog_cmd₩₩" + vs);
21. }
```

Line 10: Make sure it is of the intended type and use the value selected in the pre-generated version array based on the input.Before generating the command with the external input props.getProperty ("dir_type")

Issue ID	274580
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00006.java
Line	66

Source Code

```
61.         }
62.         argList.add("echo " + param);
63.
64.         ProcessBuilder pb = new ProcessBuilder();
65.
66.         pb.command(argList);
67.
68.         try {
69.             Process p = pb.start();
70.             org.owasp.benchmark.helpers.Utils.printOSCommandResults(p, response);
71.         } catch (IOException e) {
```

Issue ID	274581
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00007.java
Line	61

Source Code

```
56.        String[] argsEnv = {param};
57.
58.        Runtime r = Runtime.getRuntime();
59.
60.        try {
61.            Process p = r.exec(args, argsEnv);
62.            org.owasp.benchmark.helpers.Utils.printOSCommandResults(p, response);
63.        } catch (IOException e) {
64.            System.out.println("Problem executing cmdi - TestCase");
65.            response.getWriter()
66.                    .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(e.getMessage()));
```

Issue ID	274592
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00017.java
Line	63

Source Code

```
58.        }
59.
60.        Runtime r = Runtime.getRuntime();
61.
62.        try {
63.            Process p = r.exec(cmd + param);
64.            org.owasp.benchmark.helpers.Utils.printOSCommandResults(p, response);
```

```
65.        } catch (IOException e) {
66.            System.out.println("Problem executing cmdi - TestCase");
67.            response.getWriter()
68.                    .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(e.
getMessage()));
```

Issue ID	274669
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00077.java
Line	98

Source Code

```
93.            argList.add("sh");
94.            argList.add("-c");
95.        }
96.        argList.add("echo " + bar);
97.
98.        ProcessBuilder pb = new ProcessBuilder(argList);
99.
100.        try {
101.            Process p = pb.start();
102.            org.owasp.benchmark.helpers.Utils.printOSCommandResults(p,
response);
103.        } catch (IOException e) {
```

Issue ID	274697
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00092.java
Line	95

Source Code

```
90.        String[] argsEnv = {bar};
91.
```

```
    Runtime r = Runtime.getRuntime();

    try {
        Process p = r.exec(args, argsEnv);
        org.owasp.benchmark.helpers.Utils.printOSCommandResults(p, response);
    } catch (IOException e) {
        System.out.println("Problem executing cmdi - TestCase");
        response.getWriter()
                .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(e.getMessage()));
```

● [Rule Name] LDAP Injection (High, Java)

The LDAP Injection checker finds LDAP (Lightweight Directory Access Protocol) queries built with an unvalidated external input.

Unvalidated external inputs can be exploited to execute unintended LDAP commands. To be specific, if the web application does not handle user inputs properly, attackers can alter LDAP query commands, thereby enabling the process to behave with the same authentication as the command executor component.

Make sure user inputs provided for distinguished names (DNs) and filters do not include special characters. If special characters have to be allowed in inputs, have =, +, <, >, #, ;, and recognized as plain characters, not as executable commands.

- 소프트웨어 보안약점 진단가이드 2021
    - LDAP 삽입

Dangerous Example

```
private void searchRecord(String userSN, String userPassword) throws NamingException {
  Hashtable<String, String> env = new Hashtable<String, String>();
  env.put(Context.INITIAL_CONTEXT_FACTORY, "com.sun.jndi.ldap.LdapCtxFactory");
  try {
    DirContext dctx = new InitialDirContext(env);
    SearchControls sc = new SearchControls();
    String[] attributeFilter = { "cn", "mail" };
8.    sc.setReturningAttributes(attributeFilter);
9.    sc.setSearchScope(SearchControls.SUBTREE_SCOPE);
10.    String base = "dc=example,dc=com";
11.    String filter = "(&(sn=" + userSN + ")(userPassword=" + userPassword + "))";
12.    NamingEnumeration<?> results = dctx.search(base, filter, sc);
13.    while (results.hasMore()) {
14.      SearchResult sr = (SearchResult) results.next();
15.      Attributes attrs = sr.getAttributes();
16.      Attribute attr = attrs.get("cn");
17.      ...
18.    }
19.    dctx.close();
20.  } catch (NamingException e) { ... }
21. }
```

Line 11: A filter string condition is always true by passing * as variables to userSN and userPassword, which can cause unintended behaviors.

Safe Example

```
1. private void searchRecord(String userSN, String userPassword) throws
NamingException {
2.  Hashtable<String, String> env = new Hashtable<String, String>();
3.  env.put(Context.INITIAL_CONTEXT_FACTORY, "com.sun.jndi.ldap.
LdapCtxFactory");
4.  try {
5.    DirContext dctx = new InitialDirContext(env);
6.    SearchControls sc = new SearchControls();
7.    String[] attributeFilter = { "cn", "mail" };
8.    sc.setReturningAttributes(attributeFilter);
9.    sc.setSearchScope(SearchControls.SUBTREE_SCOPE);
10.    String base = "dc=example,dc=com";
11.    if (!userSN.matches("[₩₩w₩₩s]*") || !userPassword.matches("[₩₩w]*")) {
12.      throw new IllegalArgumentException("Invalid input");
13.    }
14.    String filter = "(&(sn=" + userSN + ")(userPassword=" + userPassword + "))";
15.    NamingEnumeration<?> results = dctx.search(base, filter, sc);
16.    while (results.hasMore()) {
17.      SearchResult sr = (SearchResult) results.next();
18.      Attributes attrs = sr.getAttributes();
19.      Attribute attr = attrs.get("cn");
20.      ...
21.    }
22.    dctx.close();
23.  } catch (NamingException e) { ... }
24. }
```

Line 11: Remove tainted strings from external inputs used as the filter string for search, which can partially reduce risks.

Issue ID	274588
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00012.java
Line	68

Source Code

```
63.            javax.naming.directory.DirContext ctx = ads.getDirContext();
64.            javax.naming.directory.InitialDirContext idc =
65.                    (javax.naming.directory.InitialDirContext) ctx;
66.            boolean found = false;
67.            javax.naming.NamingEnumeration<javax.naming.directory.SearchResult> results =
68.                    idc.search(base, filter, filters, sc);
69.            while (results.hasMore()) {
70.                javax.naming.directory.SearchResult sr =
71.                        (javax.naming.directory.SearchResult) results.next();
72.                javax.naming.directory.Attributes attrs = sr.getAttributes();
73.
```

Issue ID	274597
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00021.java
Line	59

Source Code

```
54.         String filter = "(&(objectclass=person))(|(uid=" + param + ")(street={0}))";
55.         Object[] filters = new Object[] {"The streetz 4 Ms bar"};
56.         // System.out.println("Filter " + filter);
57.         boolean found = false;
58.         javax.naming.NamingEnumeration<javax.naming.directory.SearchResult> results =
59.                 ctx.search(base, filter, filters, sc);
60.         while (results.hasMore()) {
61.             javax.naming.directory.SearchResult sr =
62.                     (javax.naming.directory.SearchResult) results.next();
63.             javax.naming.directory.Attributes attrs = sr.getAttributes();
64.
```

Issue ID	274623
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00044.java
Line	60

Source Code

```
55.         sc.setSearchScope(javax.naming.directory.SearchControls.SUBTREE_SCOPE);
56.         String filter = "(&(objectclass=person)(uid=" + param + "))";
57.         // System.out.println("Filter " + filter);
58.         boolean found = false;
59.         javax.naming.NamingEnumeration<javax.naming.directory.SearchResult> results =
60.                 ctx.search(base, filter, sc);
61.         while (results.hasMore()) {
62.             javax.naming.directory.SearchResult sr =
63.                     (javax.naming.directory.SearchResult) results.next();
64.             javax.naming.directory.Attributes attrs = sr.getAttributes();
65.
```

● [Rule Name] Resource injection (High, Java)

The Resource injection checker finds resource identifiers created with an unvalidated external input.

If unvalidated external inputs are allowed for accessing or identifying files, servers, or other system resources, attackers can manipulate an input to arbitrarily access system-protected resources. Resource injection vulnerabilities can be exploited to modify or delete resources, leak system information, and cause conflicts between system resources leading to service failures.

If external inputs need to be used for resource identifiers (such as socket ports), whitelist appropriate identifiers and restrict values determined by external inputs to those selected from within the whitelist.

- 무기체계 소프트웨어 보안약점 점검 목록
    - CWE-99

- 소프트웨어 보안약점 진단가이드 2021
    - 경로 조작 및 자원 삽입

Dangerous Example

```
public void foo() {
  ServerSocket serverSocket;
  Properties props = new Properties();
  String filename = "file_list";
  FileInputStream in = new FileInputStream(fileName);
  props.load(in);
  String service = props.getProperty("Service No");
  int port = Integer.parseInt(service);
  // if wrong service value is used, crash with
  // port number
  if(port != 0)
    serverSocket = new ServerSocket(port);
  else
    serverSocket = new ServerSocket(4000);
}
```

Line 12: A socket number received from outside is used without validation. If an attacker set the number to 80 and another service is already running on the socket, an error can occur by conflicts with an earlier service.

Safe Example

```
1. public void foo() {
2.   ServerSocket serverSocket;
3.   Properties props = new Properties();
4.   String filename = "file_list";
5.   FileInputStream in = new FileInputStream(fileName);
6.   String service = "";
7.   if(in != null && in.available() > 0) {
8.     props.load(in);
9.     service = props.getProperty("Service No");
10.  }
11.   if("".equals(service)) service= "8080";
12.   int port = Integer.parseInt(service);
13.   switch(port) {
14.   case 1:
15.     port = 3001; break;
16.   case 2:
17.     port = 3002; break;
18.   case 3:
19.     port = 3003; break;
20.   default:
21.     port = 3003;
22.  }
23.   serverSocket = new ServerSocket(port);
24. }
```

Line 13: Choose a predetermined port based on external input to avoid an attacker using a random port.

Issue ID	274563
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	71

Source Code

```
66.        String fileName = null;
67.        java.io.FileInputStream fis = null;
68.
69.        try {
70.            fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
71.            fis = new java.io.FileInputStream(new java.io.File(fileName));
72.            byte[] b = new byte[1000];
73.            int size = fis.read(b);
74.            response.getWriter()
75.                    .println(
76.                            "The beginning of file: '"
```

Issue ID	274566
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	71

Source Code

```
66.        String fileName = null;
67.        java.io.FileInputStream fis = null;
68.
69.        try {
70.            fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
71.            fis = new java.io.FileInputStream(new java.io.File(fileName));
72.            byte[] b = new byte[1000];
73.            int size = fis.read(b);
74.            response.getWriter()
75.                    .println(
76.                            "The beginning of file: '"
```

Issue ID	274571
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode

File	/BenchmarkTest00002.java
Line	72

Source Code

```
67.         java.io.FileOutputStream fos = null;
68.
69.         try {
70.             fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
71.
72.             fos = new java.io.FileOutputStream(fileName, false);
73.             response.getWriter()
74.                     .println(
75.                             "Now ready to write to file: "
76.                                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML
(fileName));
77.
```

Issue ID	274586
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00011.java
Line	54

Source Code

```
49.         }
50.
51.         // URL Decode the header value since req.getHeaders() doesn't. Unlike req.
getParameters().
52.         param = java.net.URLDecoder.decode(param, "UTF-8");
53.
54.         java.io.File fileTarget = new java.io.File(param, "/Test.txt");
55.         response.getWriter()
56.                 .println(
57.                         "Access to file: '"
58.                                 + org.owasp
59.                                         .esapi
```

Issue ID	274610
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00028.java
Line	57

Source Code

```
52.         java.io.FileOutputStream fos = null;
53.
54.         try {
55.             fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
56.
57.             fos = new java.io.FileOutputStream(fileName, false);
58.             response.getWriter()
59.                     .println(
60.                             "Now ready to write to file: "
61.                                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML(fileName));
62.
```

Issue ID	274628
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00061.java
Line	74

Source Code

```
69.                             org.apache.commons.codec.binary.Base64.decodeBase64(
70.                                     org.apache.commons.codec.binary.Base64.encodeBase64(
71.                                             param.getBytes())));
72.         }
73.
74.         java.io.File fileTarget = new java.io.File(bar, "/Test.txt");
75.         response.getWriter()
76.                 .println(
77.                     "Access to file: '"
78.                             + org.owasp
79.                                     .esapi
```

Issue ID	274635
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00062.java
Line	77

Source Code

```
72.         String fileName = null;
73.         java.io.FileInputStream fis = null;
74.
75.         try {
76.             fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + bar;
77.             fis = new java.io.FileInputStream(new java.io.File(fileName));
78.             byte[] b = new byte[1000];
79.             int size = fis.read(b);
80.             response.getWriter()
81.                     .println(
82.                             "The beginning of file: '"
```

Issue ID	274636
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00062.java
Line	77

Source Code

```
72.         String fileName = null;
73.         java.io.FileInputStream fis = null;
74.
75.         try {
76.             fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + bar;
```

```
77.            fis = new java.io.FileInputStream(new java.io.File(fileName));
78.            byte[] b = new byte[1000];
79.            int size = fis.read(b);
80.            response.getWriter()
81.                    .println(
82.                            "The beginning of file: '"
```

Issue ID	274645
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00065.java
Line	79

Source Code

```
74.        String fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + bar;
75.        java.io.InputStream is = null;
76.
77.        try {
78.            java.nio.file.Path path = java.nio.file.Paths.get(fileName);
79.            is = java.nio.file.Files.newInputStream(path, java.nio.file.
StandardOpenOption.READ);
80.            byte[] b = new byte[1000];
81.            int size = is.read(b);
82.            response.getWriter()
83.                    .println(
84.                            "The beginning of file: '"
```

● [Rule Name] Path Traversal (High, Java)

The Path Manipulation checker finds instances in which an unvalidated external input creates a path accessible to the file system.

If unvalidated external inputs are allowed to be used for access to files, servers, or other system resources, attackers can manipulate inputs to access an unintended path in the program. In other words, path manipulation can be used to obtain illegitimate access to modify or execute settings files.

If external inputs are used as identifiers for resources (such as files), make sure they pass appropriate validation. Especially, if external inputs are filenames, use a filter to remove characters such as ", /, , and .. that can be exploited for a directory traversal attack.

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-22

- 소프트웨어 보안약점 진단가이드 2021
  - 경로 조작 및 자원 삽입

Dangerous Example

```
1. public void foo(Properties request){
2.   String name = request.getProperty("filename");
3.   if(name!=null){
4.     File file = new File("/usr/local/tmp/" + name);
5.     // if other file names comes into name, harmful
6.     file.delete();
7.   }
8. }
```

Line 4: An external input request.getProperty("filename") is used to access to a file without validation.

Safe Example

```
public void foo(Properties request){
  String name = request.getProperty("filename");
  if(name!=null && !"".equeals(name)){
    name = name.replaceAll("/","");
    name = name.replaceAll("₩₩","");
    name = name.replaceAll(".","");
    name = name.replaceAll("&","");
    name = name + "-report";
    File file = new file("/usr/local/tmp/" + name);
    if(file != null)  file.delete();
  }
}
```

Line 4: Use a filter to remove special characters vulnerable to attacks before accessing to the file with the external input.

Issue ID	274565
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	71

Source Code

```
66.    String fileName = null;
67.    java.io.FileInputStream fis = null;
68.
69.    try {
70.        fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
71.        fis = new java.io.FileInputStream(new java.io.File(fileName));
72.        byte[] b = new byte[1000];
73.        int size = fis.read(b);
74.        response.getWriter()
75.                .println(
76.                        "The beginning of file: '"
```

Issue ID	274567
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	71

Source Code

```
66.    String fileName = null;
67.    java.io.FileInputStream fis = null;
68.
69.    try {
70.        fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
```

```
71.          fis = new java.io.FileInputStream(new java.io.File(fileName));
72.          byte[] b = new byte[1000];
73.          int size = fis.read(b);
74.          response.getWriter()
75.                  .println(
76.                          "The beginning of file: '"
```

Issue ID 274570

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00002.java

Line 72

Source Code

```
67.      java.io.FileOutputStream fos = null;
68.
69.      try {
70.          fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
71.
72.          fos = new java.io.FileOutputStream(fileName, false);
73.          response.getWriter()
74.                  .println(
75.                          "Now ready to write to file: "
76.                                  + org.owasp.esapi.ESAPI.encoder().encodeForHTML(fileName));
77.
```

Issue ID 274587

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00011.java

Line 54

Source Code

```
49.         }
50.
51.         // URL Decode the header value since req.getHeaders() doesn't. Unlike req.getParameters().
52.         param = java.net.URLDecoder.decode(param, "UTF-8");
53.
54.         java.io.File fileTarget = new java.io.File(param, "/Test.txt");
55.         response.getWriter()
56.                 .println(
57.                         "Access to file: '"
58.                                 + org.owasp
59.                                         .esapi
```

Issue ID 274609

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00028.java

Line 57

Source Code

```
52.         java.io.FileOutputStream fos = null;
53.
54.         try {
55.             fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
56.
57.             fos = new java.io.FileOutputStream(fileName, false);
58.             response.getWriter()
59.                     .println(
60.                             "Now ready to write to file: "
61.                                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML(fileName));
62.
```

Issue ID 274627

BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode

File	/BenchmarkTest00061.java
Line	74

Source Code

```
69.           org.apache.commons.codec.binary.Base64.decodeBase64(
70.               org.apache.commons.codec.binary.Base64.encodeBase64(
71.                   param.getBytes())));
72. }
73.
74. java.io.File fileTarget = new java.io.File(bar, "/Test.txt");
75. response.getWriter()
76.     .println(
77.         "Access to file: '"
78.             + org.owasp
79.                 .esapi
```

Issue ID	274633
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00062.java
Line	77

Source Code

```
72. String fileName = null;
73. java.io.FileInputStream fis = null;
74.
75. try {
76.   fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + bar;
77.   fis = new java.io.FileInputStream(new java.io.File(fileName));
78.   byte[] b = new byte[1000];
79.   int size = fis.read(b);
80.   response.getWriter()
81.       .println(
82.           "The beginning of file: '"
```

Issue ID	274634
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00062.java
Line	77

Source Code

```
72.        String fileName = null;
73.        java.io.FileInputStream fis = null;
74.
75.        try {
76.            fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + bar;
77.            fis = new java.io.FileInputStream(new java.io.File(fileName));
78.            byte[] b = new byte[1000];
79.            int size = fis.read(b);
80.            response.getWriter()
81.                    .println(
82.                            "The beginning of file: '"
```

Issue ID	274644
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00065.java
Line	79

Source Code

```
74.        String fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + bar;
75.        java.io.InputStream is = null;
76.
77.        try {
78.            java.nio.file.Path path = java.nio.file.Paths.get(fileName);
79.            is = java.nio.file.Files.newInputStream(path, java.nio.file.
StandardOpenOption.READ);
80.            byte[] b = new byte[1000];
81.            int size = is.read(b);
```

```
        response.getWriter()
            .println(
                "The beginning of file: '"
```

● [Rule Name] HTTP Response Splitting (Medium, Java)

The HTTP Response Splitting checker finds HTTP responses created with an unvalidated external input.

When arguments passed in an HTTP request are sent back through an HTTP response header, newline characters such as CRs (carriage returns) and LFs (line feeds) contained in the inputs can split the HTTP response into two or more. By exploiting this, attackers can use a newline to stop the first response and inject a malicious code into the second one for an XSS or cache poisoning attack.

If you need to include request parameters in an HTTP response header such as Set-Cookie, make sure all CRs and LFs are filtered out.

- 소프트웨어 보안약점 진단가이드 2021
    - HTTP 응답분할

Dangerous Example

```
1. ...
2. String lastLogin = request.getParameter("last_login");
3. if (lastLogin == null || "".equals(lastLogin)) {
4.   return;
5. }
6. Cookie c = new Cookie("LASTLOGIN", lastLogin);
7. c.setMaxAge(1000);
8. c.setSecure(true);
9. response.addCookie(c);
10. response.setContentType("text/html");
11. ...
```

Line 6: A cookie value is set with an external input, lastLogin. If an attacker set the lastLogin to "Wiley Hacker/r/nHTTP/1.1 200 OK/r/n", then a detached response can be sent and its texts can be manipulated.

Safe Example

```
1. ...
2. String lastLogin = request.getParameter("last_login");
3. if (lastLogin == null || "".equals(lastLogin)) {
4.   return;
5. }
6. lastLogin = lastLogin.replaceAll("[₩₩r₩₩n]", "");
7. Cookie c = new Cookie("LASTLOGIN", lastLogin);
8. c.setMaxAge(1000);
9. c.setSecure(true);
10. response.addCookie(c);
11. response.setContentType("text/html");
12. ...
```

Line 6: To prevent a response from splitting up, remove a newline character and use it as a response header value.

Issue ID	274607
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00023.java
Line	83

Source Code

```
78.         rememberMe.setHttpOnly(true);
79.         rememberMe.setDomain(new java.net.URL(request.getRequestURL().
toString()).getHost());
80.         rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST
this servlet
81.         // e.g., /benchmark/sql-01/BenchmarkTest01001
82.         request.getSession().setAttribute(cookieName, rememberMeKey);
83.         response.addCookie(rememberMe);
84.         response.getWriter()
85.                 .println(
86.                         user
87.                                 + " has been remembered with cookie: "
88.                                 + rememberMe.getName()
```

Issue ID	274617
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00042.java
Line	86

Source Code

```
81.         rememberMe.setSecure(true);
82.         rememberMe.setHttpOnly(true);
83.         rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
84.         // e.g., /benchmark/sql-01/BenchmarkTest01001
85.         request.getSession().setAttribute(cookieName, rememberMeKey);
86.         response.addCookie(rememberMe);
87.         response.getWriter()
88.                 .println(
89.                         user
90.                                 + " has been remembered with cookie: "
91.                                 + rememberMe.getName()
```

Issue ID	274649
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00067.java
Line	125

Source Code

```
120.        rememberMe.setHttpOnly(true);
121.        rememberMe.setDomain(new java.net.URL(request.getRequestURL().toString()).getHost());
122.        rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
123.        // e.g., /benchmark/sql-01/BenchmarkTest01001
124.        request.getSession().setAttribute(cookieName, rememberMeKey);
125.        response.addCookie(rememberMe);
```

```
126.        response.getWriter()
127.                .println(
128.                        user
129.                                + " has been remembered with cookie: "
130.                                + rememberMe.getName()
```

Issue ID	274674
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00084.java
Line	104

Source Code

```
99.        rememberMe.setSecure(true);
100.        rememberMe.setHttpOnly(true);
101.        rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
102.        // e.g., /benchmark/sql-01/BenchmarkTest01001
103.        request.getSession().setAttribute(cookieName, rememberMeKey);
104.        response.addCookie(rememberMe);
105.        response.getWriter()
106.                .println(
107.                        user
108.                                + " has been remembered with cookie: "
109.                                + rememberMe.getName()
```

Issue ID	274682
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00086.java
Line	108

Source Code

```
103.        rememberMe.setSecure(true);
104.        rememberMe.setHttpOnly(true);
```

```
105.          rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST
this servlet
106.          // e.g., /benchmark/sql-01/BenchmarkTest01001
107.          request.getSession().setAttribute(cookieName, rememberMeKey);
108.          response.addCookie(rememberMe);
109.          response.getWriter()
110.                  .println(
111.                          user
112.                                  + " has been remembered with cookie: "
113.                                  + rememberMe.getName()
```

Issue ID	274694
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00087.java
Line	89

Source Code

```
84.                 return;
85.             }
86.             str = new String(input, 0, i);
87.         }
88.         if ("".equals(str)) str = "No cookie value supplied";
89.         javax.servlet.http.Cookie cookie = new javax.servlet.http.Cookie
("SomeCookie", str);
90.
91.         cookie.setSecure(false);
92.         cookie.setHttpOnly(true);
93.         cookie.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
94.         // e.g., /benchmark/sql-01/BenchmarkTest01001
```

Issue ID	274693
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00087.java
Line	95

Source Code

```

        cookie.setSecure(false);
        cookie.setHttpOnly(true);
        cookie.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
        // e.g., /benchmark/sql-01/BenchmarkTest01001
        response.addCookie(cookie);

        response.getWriter()
                .println(
                        "Created cookie: 'SomeCookie': with value: '"
                                + org.owasp.esapi.ESAPI.encoder().encodeForHTML(str)
```

● [Rule Name] XPath Injection (High, Java)

The XPath Injection checker finds XPath queries that include an unvalidated external input.

If external inputs are allowed in XPath queries without appropriately validating them, attackers can feed an unexpected string to twist a query''s meaning or alter its structure, thereby gaining access to restricted data.

External inputs used in XPath queries should be filtered for special characters (", [,], /, =, @, etc.) and reserved worlds.

- OWASP 2017
    - A1-Injection

- OWASP 2021
    - A03 Injection

- 소프트웨어 보안약점 진단가이드 2021
    - XML 삽입

Dangerous Example

```
1. String nm = props.getProperty("name");
2. String pw = props.getProperty("password");
3. ...
4. XPathFactory factory = XPathFactory.newInstance();
5. XPath xpath = factory.newXPath();
6. ...
7. XPathExpression expr = xpath.compile("//users/user[login/text()='"+nm+"' and password/text()='"+pw+"']/home_dir/text()");
8. Object result = expr.evaluate(doc, XPathConstants.NODESET);
9. NodeList nodes = (NodeList) result;
10. for (int i=0; i<nodes.getLength(); i++) {
11.   String value = nodes.item(i).getNodeValue();
12.   if (value.indexOf(">") < 0) {
13.     ...
14.   }
15. }
16.
17. public static void main(String[] args) throws Exception {
18.   ...
19.   String name = args[0];
20.   DocumentBuilder docBuilder = DocumentBuilderFactory.newInstance().newDocumentBuilder();
21.   Document doc = docBuilder.parse("http://www.w3schools.com/xml/simple.xml");
22.   XPath xpath = XPathFactory.newInstance().newXPath();
23.   NodeList nodes = (NodeList) xpath.evaluate("//food[name='" + name + "']/price", doc, XPathConstants.NODESET);
24.   for (int i = 0; i < nodes.getLength(); i++) {
25.     System.out.println(nodes.item(i).getTextContent());
26.   }
27. }
```

Line 7: An XPath query is generated without validation for name and password inputs. If the input is used to generate or execute XPath query, an attacker can manipulate the XPath query.

Safe Example

```
1. declare variable $loginID as xs:string external;
2. declare variable $password as xs:string external;
```

```
3. //users/user[@loginID=$loginID and @password=$password]
4. String nm = props.getProperty("name");
5. String pw = props.getProperty("password");
6. Document doc = new Builder().build("users.xml");
7. XQuery xquery = new XQueryFactory().createXQuery(new File("login.xq"));
8. Map vars = new HashMap();
9. vars.put("loginID", nm);
10. vars.put("password", pw);
11. Nodes results = xquery.execute(doc, null, vars).toNodes();
12. for (int i=0; i<results.size(); i++) {
13.   System.out.println(results.get(i).toXML());
14. }
15.
16. public static void main(String[] args) throws Exception {
17.   ...
18.   String name = args[0];
19.   if (name != null) {
20.     name = name.replaceAll("[()₩₩-'₩₩[₩₩]:,*/]", "");
21.   }
22.   DocumentBuilder docBuilder = DocumentBuilderFactory.newInstance().newDocumentBuilder();
23.   Document doc = docBuilder.parse("http://www.w3schools.com/xml/simple.xml");
24.   XPath xpath = XPathFactory.newInstance().newXPath();
25.   NodeList nodes = (NodeList) xpath.evaluate("//food[name='" + name + "']/price", doc, XPathConstants.NODESET);
26.   for (int i = 0; i < nodes.getLength(); i++) {
27.     System.out.println(nodes.item(i).getTextContent());
28.   }
29. }
```

Line 7: Use an XQuery to generate a query skeleton, which can prevent its query structure from being changed by the external input. Remove code to manipulate the XPath query.

Issue ID	274701
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00207.java
Line	75

Source Code

```
70.         org.w3c.dom.Document xmlDocument = builder.parse(file);
71.         javax.xml.xpath.XPathFactory xpf = javax.xml.xpath.XPathFactory.newInstance();
72.         javax.xml.xpath.XPath xp = xpf.newXPath();
73.
74.         String expression = "/Employees/Employee[@emplid='" + bar + "']";
75.         String result = xp.evaluate(expression, xmlDocument);
76.
77.         response.getWriter().println("Your query results are: " + result + "<br/>");
78.
79.     } catch (javax.xml.xpath.XPathExpressionException
80.             | javax.xml.parsers.ParserConfigurationException
```

Issue ID	274703
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00442.java
Line	68

Source Code

```
63.         org.w3c.dom.Document xmlDocument = builder.parse(file);
64.         javax.xml.xpath.XPathFactory xpf = javax.xml.xpath.XPathFactory.newInstance();
65.         javax.xml.xpath.XPath xp = xpf.newXPath();
66.
67.         String expression = "/Employees/Employee[@emplid='" + bar + "']";
68.         String result = xp.evaluate(expression, xmlDocument);
69.
70.         response.getWriter().println("Your query results are: " + result + "<br/>");
71.
72.     } catch (javax.xml.xpath.XPathExpressionException
73.             | javax.xml.parsers.ParserConfigurationException
```

● [Rule Name] Cross-site scripting (Medium, Java)

The Cross-site scripting checker finds instances of including an external input in HTML without validating it.

If unvalidated external inputs are allowed for page generation, attackers can inject malicious scripts into the pages. This vulnerability lets attackers take over a user''s cookie, session, or other information or execute an abnormal function.

To prevent external inputs from being used for scripts, use a text replacement function or method to substitute characters such as <, >, &, and " with &lt;, &gt;, &amp;, and &quot;, respectively. For bulletin boards that allow using HTML tags, make a whitelist of allowable HTML tags.

- OWASP 2021
    - A03 Injection

- 소프트웨어 보안약점 진단가이드 2021
    - 크로스사이트 스크립트

Dangerous Example

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body>
    <h1>XSS Sample</h1>
    <%
    <!- Receive name from external source -->
      String name = request.getParameter("name");
    %>
    <!?Print name received from outer source -->
    <p>NAME:<%=name%></p>
  </body>
</html>

<% String customerID = request.getParameter("id"); %>
```

```
18.
19.
```

Line 13: name with an external input is used to generate a result page without validation. If the following script is entered to name value, an attacker can execute attack.jsp with an elevated privilege to give damages such as cookie information exposure.
(For example : <script>URL = "http://devil.com/attack.jsp";</script>)
Line 17: A parameter ID is entered with script code that prints out cookie information. Then, attackers can use code to steal the cookie information.

Safe Example

```
1. <%@page contentType="text/html" pageEncoding="UTF-8"%>
2. <html>
3.   <head>
4.     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
5.   </head>
6.   <body>
7.     <h1>XSS Sample</h1>
8.     <%
9.     <!- Receive name from outer source -->
10.    String name = request.getParameter("name");
11.     <!? Print name received from outer source -->
12.     if ( name != null ) {
13.       name = name.replaceAll("<","&lt;");
14.       name = name.replaceAll(">","&gt;");
15.     } else {
16.       return;
17.     }
18.     %>
19.     <!-- Remove dangerous character from name received from
20.          outer source, then print it -->
21.     <p>NAME:<%=name%></p>
22.   </body>
23. </html>
24.
25. <textarea name="content">${ fn:escapeXml(model.content) }</textarea>
26. ...
27. <textarea name="content"><c:out value="${model.content}"/></textarea>
28. ...
29. XssFilter filter = XssFilter.getInstance("lucy-xss-superset.xml");
```

```
30. out.append(filter.doFilter(data));
31.
```

Line 13: Use replaceAll() method to change "<" and ">" used for HTML scripts into "&lt;" and "&gt;", which can reduce risks to run malicious scripts. But this cannot completely remove the risks. There are other ways to prevent the attacks.
Line 25: Encode outputs with JSTL HTML in JSP.
Line 27: the output is treated as a text by using a JSTL Core output format in JSP.
Line 30: Filter the output by using a well-designed an external XSSFilter library.

Issue ID	274576
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00003.java
Line	101

Source Code

```
96.                  "hash_value="
97.                          + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
98.                          + "₩n");
99.          fw.close();
100.         response.getWriter()
101.                 .println(
102.                         "Sensitive value '"
103.                                 + org.owasp
104.                                         .esapi
105.                                         .ESAPI
106.                                         .encoder()
```

Issue ID	274579
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00005.java
Line	100

Source Code

```
95.                 "secret_value="
96.                         + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
97.                         + "₩n");
98.         fw.close();
99.         response.getWriter()
100.                .println(
101.                        "Sensitive value: '"
102.                                + org.owasp
103.                                        .esapi
104.                                        .ESAPI
105.                                        .encoder()
```

Issue ID	274585
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00009.java
Line	101

Source Code

```
96.                 "hash_value="
97.                         + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
98.                         + "₩n");
99.         fw.close();
100.          response.getWriter()
101.                .println(
102.                        "Sensitive value '"
103.                                + org.owasp
104.                                        .esapi
105.                                        .ESAPI
106.                                        .encoder()
```

Issue ID	274590
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00012.java

Line	78

Source Code

```
73.
74.            javax.naming.directory.Attribute attr = attrs.get("uid");
75.            javax.naming.directory.Attribute attr2 = attrs.get("street");
76.            if (attr != null) {
77.              response.getWriter()
78.                  .println(
79.                      "LDAP query results:<br>"
80.                          + "Record found with name "
81.                          + attr.get()
82.                          + "<br>"
83.                          + "Address: "
```

Issue ID	274595
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00020.java
Line	95

Source Code

```
90.              "secret_value="
91.                  + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
92.                  + "₩n");
93.          fw.close();
94.          response.getWriter()
95.              .println(
96.                  "Sensitive value: '"
97.                      + org.owasp
98.                          .esapi
99.                          .ESAPI
100.                          .encoder()
```

Issue ID	274600

File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00021.java
Line	69

Source Code

```
64.
65.         javax.naming.directory.Attribute attr = attrs.get("uid");
66.         javax.naming.directory.Attribute attr2 = attrs.get("street");
67.         if (attr != null) {
68.           response.getWriter()
69.               .println(
70.                   "LDAP query results:<br>"
71.                       + "Record found with name "
72.                       + attr.get()
73.                       + "<br>"
74.                       + "Address: "
```

Issue ID	274602
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00023.java
Line	73

Source Code

```
68.         }
69.       }
70.     }
71.
72.     if (foundUser) {
73.       response.getWriter().println("Welcome back: " + user + "<br/>");
74.     } else {
75.       javax.servlet.http.Cookie rememberMe =
76.           new javax.servlet.http.Cookie(cookieName, rememberMeKey);
77.       rememberMe.setSecure(true);
78.       rememberMe.setHttpOnly(true);
```

Issue ID	274601
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00023.java
Line	85

Source Code

```
80.        rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
81.        // e.g., /benchmark/sql-01/BenchmarkTest01001
82.        request.getSession().setAttribute(cookieName, rememberMeKey);
83.        response.addCookie(rememberMe);
84.        response.getWriter()
85.                .println(
86.                        user
87.                                + " has been remembered with cookie: "
88.                                + rememberMe.getName()
89.                                + " whose value is: "
90.                                + rememberMe.getValue()
```

Issue ID	274618
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00042.java
Line	77

Source Code

```
72.                }
73.            }
74.        }
75.
76.        if (foundUser) {
77.            response.getWriter().println("Welcome back: " + user + "<br/>");
78.        } else {
79.            javax.servlet.http.Cookie rememberMe =
80.                    new javax.servlet.http.Cookie(cookieName, rememberMeKey);
```

```
81.             rememberMe.setSecure(true);
82.             rememberMe.setHttpOnly(true);
```

Issue ID	274619
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00042.java
Line	88

Source Code

```
83.             rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
84.             // e.g., /benchmark/sql-01/BenchmarkTest01001
85.             request.getSession().setAttribute(cookieName, rememberMeKey);
86.             response.addCookie(rememberMe);
87.             response.getWriter()
88.                     .println(
89.                             user
90.                                     + " has been remembered with cookie: "
91.                                     + rememberMe.getName()
92.                                     + " whose value is: "
93.                                     + rememberMe.getValue()
```

Issue ID	274621
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00044.java
Line	70

Source Code

```
65.
66.             javax.naming.directory.Attribute attr = attrs.get("uid");
67.             javax.naming.directory.Attribute attr2 = attrs.get("street");
68.             if (attr != null) {
69.                 response.getWriter()
```

```
70.                         .println(
71.                             "LDAP query results:<br>"
72.                                     + "Record found with name "
73.                                     + attr.get()
74.                                     + "<br>"
75.                                     + "Address: "
```

Issue ID	274651
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00067.java
Line	114

Source Code

```
109.                }
110.            }
111.        }
112.
113.        if (foundUser) {
114.            response.getWriter().println("Welcome back: " + user + "<br/>");
115.
116.        } else {
117.            javax.servlet.http.Cookie rememberMe =
118.                    new javax.servlet.http.Cookie(cookieName, rememberMeKey);
119.            rememberMe.setSecure(true);
```

Issue ID	274654
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00067.java
Line	127

Source Code

```
122.            rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
```

```
123.        // e.g., /benchmark/sql-01/BenchmarkTest01001
124.        request.getSession().setAttribute(cookieName, rememberMeKey);
125.        response.addCookie(rememberMe);
126.        response.getWriter()
127.                .println(
128.                        user
129.                                + " has been remembered with cookie: "
130.                                + rememberMe.getName()
131.                                + " whose value is: "
132.                                + rememberMe.getValue()
```

Issue ID	274661
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00070.java
Line	103

Source Code

```
98.               "hash_value="
99.                       + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
100.                        + "₩n");
101.        fw.close();
102.        response.getWriter()
103.                .println(
104.                        "Sensitive value '"
105.                                + org.owasp
106.                                        .esapi
107.                                        .ESAPI
108.                                        .encoder()
```

Issue ID	274663
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00076.java
Line	121

Source Code

```
116.                "hash_value="
117.                    + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result,
true)
118.                    + "₩n");
119.        fw.close();
120.        response.getWriter()
121.                .println(
122.                    "Sensitive value '"
123.                        + org.owasp
124.                            .esapi
125.                            .ESAPI
126.                            .encoder()
```

Issue ID	274679
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00084.java
Line	95

Source Code

```
90.         }
91.       }
92.     }
93.
94.     if (foundUser) {
95.       response.getWriter().println("Welcome back: " + user + "<br/>");
96.     } else {
97.       javax.servlet.http.Cookie rememberMe =
98.           new javax.servlet.http.Cookie(cookieName, rememberMeKey);
99.       rememberMe.setSecure(true);
100.      rememberMe.setHttpOnly(true);
```

Issue ID	274673
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode

File /BenchmarkTest00084.java

Line 106

Source Code

```
101.          rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
102.          // e.g., /benchmark/sql-01/BenchmarkTest01001
103.          request.getSession().setAttribute(cookieName, rememberMeKey);
104.          response.addCookie(rememberMe);
105.          response.getWriter()
106.                  .println(
107.                          user
108.                                  + " has been remembered with cookie: "
109.                                  + rememberMe.getName()
110.                                  + " whose value is: "
111.                                  + rememberMe.getValue()
```

Issue ID 274681

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00086.java

Line 98

Source Code

```
93.                 }
94.             }
95.         }
96.
97.         if (foundUser) {
98.             response.getWriter().println("Welcome back: " + user + "<br/>");
99.
100.        } else {
101.            javax.servlet.http.Cookie rememberMe =
102.                    new javax.servlet.http.Cookie(cookieName, rememberMeKey);
103.            rememberMe.setSecure(true);
```

Issue ID	274685
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00086.java
Line	110

Source Code

```
105.        rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
106.        // e.g., /benchmark/sql-01/BenchmarkTest01001
107.        request.getSession().setAttribute(cookieName, rememberMeKey);
108.        response.addCookie(rememberMe);
109.        response.getWriter()
110.                .println(
111.                        user
112.                                + " has been remembered with cookie: "
113.                                + rememberMe.getName()
114.                                + " whose value is: "
115.                                + rememberMe.getValue()
```

Issue ID	274699
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00207.java
Line	77

Source Code

```
72.         javax.xml.xpath.XPath xp = xpf.newXPath();
73.
74.         String expression = "/Employees/Employee[@emplid='" + bar + "']";
75.         String result = xp.evaluate(expression, xmlDocument);
76.
77.         response.getWriter().println("Your query results are: " + result + "<br/>");
78.
79.     } catch (javax.xml.xpath.XPathExpressionException
80.             | javax.xml.parsers.ParserConfigurationException
```

```
81.             | org.xml.sax.SAXException e) {
82.         response.getWriter()
```

Issue ID 274702

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00207.java

Line 83

Source Code

```
78.
79.     } catch (javax.xml.xpath.XPathExpressionException
80.             | javax.xml.parsers.ParserConfigurationException
81.             | org.xml.sax.SAXException e) {
82.         response.getWriter()
83.                 .println(
84.                         "Error parsing XPath input: '"
85.                                 + org.owasp.esapi.ESAPI.encoder().encodeForHTML(bar)
86.                                 + "'");
87.         throw new ServletException(e);
88.     }
```

Issue ID 274704

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00442.java

Line 70

Source Code

```
65.         javax.xml.xpath.XPath xp = xpf.newXPath();
66.
67.         String expression = "/Employees/Employee[@emplid='" + bar + "']";
68.         String result = xp.evaluate(expression, xmlDocument);
69.
70.         response.getWriter().println("Your query results are: " + result + "<br/>");
```

```

    } catch (javax.xml.xpath.XPathExpressionException
            | javax.xml.parsers.ParserConfigurationException
            | org.xml.sax.SAXException e) {
        response.getWriter()
```

[Rule Name] Improper random number generation (High, Java)

The Improper random number generation checker finds code with predictable random values.

You can optionally specify random number generating methods that should not be used.

If a predictable random number is used when an unpredictable number is demanded, an attacker can predict subsequently generated numbers and use them to attack the system.

Use secure way to generate random numbers rather than forbidden methods specified in the options.

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-330

- 소프트웨어 보안약점 진단가이드 2021
  - 적절하지 않은 난수 값 사용

Dangerous Example

```
import java.Math;
...
public static int[] insertRandom(int[] Cnt, inti, int scope) {
  int ran = (int) (Math.random() * scope) - 1;
  if (checkDigit(ran, Cnt)) {
    Cnt[i] = ran;
  } else {
    insertRandom(Cnt, i, scope);
  }
  return Cnt;
}
```

Line 4: Seeds cannot be reset by Random() method of java.lang.Math class, which is insecure.

Safe Example

```
1. import java.util.Random;
2. ...
3. public static int[] insertRandom(int[] Cnt, inti, int scope) {
4.   Random jur = new Random();
5.   jur.setSeed(new Date().getTime());
6.   int ran = (int) (jur.nextInt() * scope) - 1;
7.   if (checkDigit(ran, Cnt)) {
8.     Cnt[i] = ran;
9.   } else {
10.     insertRandom(Cnt, i, scope);
11.   }
12.   return Cnt;
13. }
```

Line 5: Use the java.util.Random class to reset the seed. Therefore, Random class can be more secure to use.

Issue ID	274533
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00067.java
Line	88

Source Code

```
83.        org.owasp.benchmark.helpers.ThingInterface thing =
84.                org.owasp.benchmark.helpers.ThingFactory.createThing();
85.        String g71153 = "barbarians_at_the_gate"; // This is static so this whole flow is 'safe'
86.        String bar = thing.doSomething(g71153); // reflection
87.
88.        double value = java.lang.Math.random();
89.        String rememberMeKey = Double.toString(value).substring(2); // Trim off the
```

```
0. at the front.

        String user = "Doug";
        String fullClassName = this.getClass().getName();
        String testCaseNumber =
```

● [Rule Name] Insecure Cookie (High, Java)

The Insecure Cookie checker finds instances of transferring an unencrypted cookie.

For browser cookies that store data, the setSecure() methods must be called with true.

You may assume that if services run solely over HTTPS, all information will be safely transferred in ciphertext.However, if security-sensitive data is stored in a browser cookie without the security attribute set enabled, it is not transferred via a security protocol such as SSL. This may expose sensitive information in plaintext to attackers. This vulnerability must be noted especially when the cookie contains privacy information or a session ID. Note that if the setSecure method is called in a site (domain) that uses HTTP only or HTTP and HTTPS together, data in browser cookies may not be transferred leading to a service failure.

Make sure that the setSecure() method is called on every cookie object with a value of true passed in.

- 소프트웨어 보안약점 진단가이드 2021
    - 암호화되지 않은 중요정보

Dangerous Example

```
private final String ACCOUNT_ID = "account";
public void setupCookies(ServletRequest r, HttpServletResponse response) {
  String acctID = r.getParameter("accountID");
  // Cookie without security attributes
  Cookie c = new Cookie(ACCOUNT_ID, acctID);
  response.addCookie(c);
}
```

Line 6: Cookie is sent in HTTPS only without security properties, which can expose information to attackers.

Safe Example

```
1. private final String ACCOUNT_ID = "account";
2. public void setupCookies(ServletRequest r, HttpServletResponse response) {
3.   String acctID = r.getParameter("accountID");
4.   // Check validity of account
5.   if (acctID == null || "".equals(acctID)) return;
6.   String filtered_ID = acctID.replaceAll("₩r", "");
7.   Cookie c = new Cookie(ACCOUNT_ID, filtered_ID);
8.   // Cookie with sensitive information need to have secure attributes.
9.   c.setSecure(true);
10.   response.addCookie(c);
11. }
```

Line 9: Call setSecure(true) method of Cookie object in case of using cookies with sensitive information sent in HTTPS only.

Issue ID	274691
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00087.java
Line	95

Source Code

```
90.
91.         cookie.setSecure(false);
92.         cookie.setHttpOnly(true);
93.         cookie.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
94.         // e.g., /benchmark/sql-01/BenchmarkTest01001
95.         response.addCookie(cookie);
96.
97.         response.getWriter()
98.                 .println(
99.                         "Created cookie: 'SomeCookie': with value: '"
100.                                 + org.owasp.esapi.ESAPI.encoder().encodeForHTML(str)
```

Issue ID	274535
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00087.java
Line	95

Source Code

```
90.
91.        cookie.setSecure(false);
92.        cookie.setHttpOnly(true);
93.        cookie.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
94.        // e.g., /benchmark/sql-01/BenchmarkTest01001
95.        response.addCookie(cookie);
96.
97.        response.getWriter()
98.                .println(
99.                        "Created cookie: 'SomeCookie': with value: '"
100.                                + org.owasp.esapi.ESAPI.encoder().encodeForHTML(str)
```

● [Rule Name] Integer Overflow (High, Java)

The Integer Overflow checker finds instances in which the result of an arithmetic operation exceeds the size of the given integer type.

If the result is out of the range of the integer type, an overflow occurs to return a negative number or an unexpected value. If such an unexpected return value is used for memory allocation or a loop conditional, the program may become vulnerable to security threats.

You must consider the ranges of integer types specific to each language or platform. When an integer-type variable is used in an operation, use a module that checks the possible range of the result value. If an external input is used for dynamic memory allocation, the variable must be checked for being within the valid range.

- CWE 660 4.14
    - 191 - Integer Underflow (Wrap or Wraparound)

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-190

- 소프트웨어 보안약점 진단가이드 2021
  - 정수형 오버플로우

Dangerous Example

```
1. public static void main(String[] args) {
2.   int size = new Integer(args[0]).intValue();
3.   size += new Integer(args[1]).intValue();
4.   MyClass[] data = new MyClass[size];
5. }
```

Line 2: A value is dynamically evaluated from an external input args[0], then used to determine the array size.
Line 4: When the array size is evaluated to a negative due to an overflow, which can cause a problem on the system.

Safe Example

```
1. public static void main(String[] args) {
2.   int size = new Integer(args[0]).intValue();
3.   size += new Integer(args[1]).intValue();
4.   // Check if size of the array is negative.
5.   if (size < 0) return ;
6.   MyClass[ ] data = new MyClass[size];
7. }
```

Line 5: Verify the size value to be used for dynamic memory assignment.

Issue ID	274653
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00067.java

Line 69

Source Code

```
64.
65.        // Chain a bunch of propagators in sequence
66.        String a71153 = param; // assign
67.        StringBuilder b71153 = new StringBuilder(a71153); // stick in stringbuilder
68.        b71153.append(" SafeStuff"); // append some safe content
69.        b71153.replace(
70.                b71153.length() - "Chars".length(),
71.                b71153.length(),
72.                "Chars"); // replace some of the end content
73.        java.util.HashMap<String, Object> map71153 = new java.util.
HashMap<String, Object>();
74.        map71153.put("key71153", b71153.toString()); // put in a collection
```

● [Rule Name] TOCTOU race condition (Medium, Java)

The TOCTOU race condition checker finds race conditions occurring between the time of checking a resource''s state and the time of using it.

When coding to check the state of a resource and access it based on the checked state, programmers often assume that the checked state will persist until the resource is accessed. But the resource''s state can change between the check and the use. This can happen even in a non-multithreaded program because the state of resources such as files can be changed by other programs or the operating system. These race conditions can cause the program to perform invalid actions or throw an exception and may help attackers gain access to unauthorized resources.

If resource accessibility needs to be determined based on resource state, use approaches such as synchronization that ensure exclusive access to resources. Or, there are some APIs available that help implement actions handling file resources. They attempt the given resource action and return different results based on the resource''s state.

- CWE 660 4.14
  - 1341 - Multiple Releases of Same Resource or Handle
  - 366 - Race Condition within a Thread

- 무기체계 소프트웨어 보안약점 점검 목록
  - CWE-367

- 소프트웨어 보안약점 진단가이드 2021
  - 경쟁조건: 검사시점과 사용시점(TOCTOU)

Dangerous Example

```
class FileMgmtThread extends Thread {
  private String manageType = "";
  public FileMgmtThread (String type) {
    manageType = type;
  }
  public void run() {
    try {
      if ( manageType.equals("READ") ) {
        File f = new File("Test_367.txt");
        if (f.exists()) { // Read contents if a file exists
          BufferedReader br = new BufferedReader(new FileReader(f));
          br.close();
        }
      } else if ( manageType.equals("DELETE") ) {
        File f = new File("Test_367.txt");
        if (f.exists()) { // delete a file if it exists
          f.delete();
        } else {... }
      }
    } catch (IOException e) {...}
  }
}
public class CWE367 {
  public static void main(String[] args) {
    // Read and delete a file simultaneously
    FileMgmtThread fileAccessThread = new FileMgmtThread("READ");
    FileMgmtThread fileDeleteThread = new FileMgmtThread("DELETE");
    fileAccessThread.start();
    fileDeleteThread.start();
30.   }
31. }
```

Line 10: A file is checked, and its result creates a branch.
Line 16: A program can be executed in an unintended way by deleting files between time-of-check and time-of-use.

Safe Example

```
1. class FileMgmtThread extends Thread {
2.        private String manageType = "";
3.        public FileMgmtThread (String type) {
4.            manageType = type;
5.        }
6.
7.        // Add synchronized for TOCTOU problem
8.        public synchronized void run() {
9.            try {
10.               if ( manageType.equals("READ") ) {
11.                   File f = new File("Test_367.txt");
12.                   // Read contents of a file if it exists
13.                   if (f.exists()) {
14.                       try {
15.                           BufferedReader br = new BufferedReader(new FileReader(f));
16.                           br.close();
17.                       } catch (FileNotFoundException e) {
18.                           // handle race condition
19.                       }
20.                   }
21.               } else if ( manageType.equals("DELETE") ) {
22.                   File f = new File("Test_367.txt");
23.                   if (f.exists()) { // Delete a file if it exists
24.                       if (f.delete()) {
25.                           // successful
26.                       } else {
27.                           // handle race condition
28.                       }
29.                   } else {...}
30.               }
31.           } catch (IOException e) {...}
32.       }
33. }
34.
35. public class CWE367 {
36.     public static void main(String[] args) {
37.         // Read and delete a file simultaneously
38.         FileMgmtThread fileAccessThread = new FileMgmtThread("READ");
39.         FileMgmtThread fileDeleteThread = new FileMgmtThread("DELETE");
40.         fileAccessThread.start();
41.         fileDeleteThread.start();
42.     }
43. }
```

Line 8: For multiple threads accessing to the shared resources, use sync statements to allow one thread to access at a time.
Line 17: The FileNotFound error will be occur in case of no file to access. Handle the exception.

Issue ID	274551
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	323

Source Code

```
318.        List<String> sourceLines = new ArrayList<String>();
319.
320.        try (FileReader fr = new FileReader(file);
321.                BufferedReader br = new BufferedReader(fr); ) {
322.            String line;
323.            while ((line = br.readLine()) != null) {
324.                sourceLines.add(line);
325.            }
326.        } catch (Exception e) {
327.            try {
328.                System.out.println("Problem reading contents of file: " + file.
getCanonicalFile());
```

● [Rule Name] Reliance on untrusted inputs in security decisions (High, Java)

The Reliance on untrusted inputs in security decisions checker finds instances of storing user credentials in a cookie.

Developers may assume that inputs such as cookies, environment variables, and hidden form fields cannot be modified. However, an attacker could change these inputs using various approaches and this change might not be detected. When security decisions such as authentication and authorization are made based on the values of these inputs, attackers can bypass the security checks of the application. Therefore, inputs from external users should not be trusted.

Make sure that critical information such as state information, sensitive data, and user sessions is stored in the server, and security checks are performed on the server side. Understand all the potential areas where untrusted inputs can enter your application. Identify all inputs that are used for security decisions and determine if you can modify the design so that you do not have to rely on submitted inputs at all.

- 소프트웨어 보안약점 진단가이드 2021
  - 보안기능 결정에 사용되는 부적절한 입력값

Dangerous Example

```
1. <%
2. String username = request.getParameter("username");
3. String password = request.getParameter("password");
4. if (username==nill || password==null || !isAuthenticatedUser(usename,
5. password)) {
6. throw new MyException("Authentification error");
7. }
8. Cookie userCookie = new Cookie("user",username);
9. Cookie authCookie = new Cookie("authenticated","1");
10. response.addCookie(userCookie);
11. response.addCookie(authCookie);
12. %>
```

Line 9: User authentication information and "authenticated" are saved to cookies in plain texts. This enables attackers to change cookie information.

Safe Example

```
1. <%
2. String username = request.getParameter("username");
3. String password = request.getParameter("password");
4. if (username==nill || password==null || !isAuthenticatedUser(usename,
5. password)) {
6.   throw new MyException("Authentication Error");
7. }
8. // Save user information in session
9. HttpSession ses = request.getSession();
10. ses.setAttribute("user",username);
11. ses.setAttribute("authenticated","1");
12. %>
```

Line 11: Save the user authentication information into sessions, which can remove risks to be exposed.

Issue ID	274548
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	167

Source Code

```
162.        Cookie[] values = request.getCookies();
163.        String param = "none";
164.        if (paramName != null) {
165.            for (int i = 0; i < values.length; i++) {
166.                if (values[i].getName().equals(paramName)) {
167.                    param = values[i].getValue();
168.                    break; // break out of for loop when param found
169.                }
170.            }
171.        }
172.        return param;
```

Issue ID	274564
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	60

Source Code

```
55.
56.     String param = "noCookieValueSupplied";
57.     if (theCookies != null) {
58.         for (javax.servlet.http.Cookie theCookie : theCookies) {
59.             if (theCookie.getName().equals("BenchmarkTest00001")) {
60.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
61.                 break;
62.             }
63.         }
64.     }
65.
```

Issue ID	274569
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00002.java
Line	60

Source Code

```
55.
56.     String param = "noCookieValueSupplied";
57.     if (theCookies != null) {
58.         for (javax.servlet.http.Cookie theCookie : theCookies) {
59.             if (theCookie.getName().equals("BenchmarkTest00002")) {
60.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
61.                 break;
62.             }
63.         }
64.     }
65.
```

Issue ID	274575
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00003.java
Line	60

Source Code

```
55.
56.     String param = "noCookieValueSupplied";
57.     if (theCookies != null) {
58.         for (javax.servlet.http.Cookie theCookie : theCookies) {
59.             if (theCookie.getName().equals("BenchmarkTest00003")) {
60.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
61.                 break;
62.             }
63.         }
64.     }
65.
```

Issue ID	274608
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00023.java
Line	65

Source Code

```
60.     javax.servlet.http.Cookie[] cookies = request.getCookies();
61.     if (cookies != null) {
62.         for (int i = 0; !foundUser && i < cookies.length; i++) {
63.             javax.servlet.http.Cookie cookie = cookies[i];
64.             if (cookieName.equals(cookie.getName())) {
```

```
65.                     if (cookie.getValue().equals(request.getSession().getAttribute(cookieName))) {
66.                         foundUser = true;
67.                     }
68.                 }
69.             }
70.         }
```

Issue ID	274604
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00023.java
Line	90

Source Code

```
85.                     .println(
86.                             user
87.                                     + " has been remembered with cookie: "
88.                                     + rememberMe.getName()
89.                                     + " whose value is: "
90.                                     + rememberMe.getValue()
91.                                     + "<br/>");
92.         }
93.
94.         response.getWriter().println("Weak Randomness Test java.util.Random.nextFloat() executed");
95.     }
```

Issue ID	274613
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00042.java
Line	68

Source Code

```
63.         javax.servlet.http.Cookie[] cookies = request.getCookies();
64.         if (cookies != null) {
65.             for (int i = 0; !foundUser && i < cookies.length; i++) {
66.                 javax.servlet.http.Cookie cookie = cookies[i];
67.                 if (cookieName.equals(cookie.getName())) {
68.                     if (cookie.getValue()
69.                             .equals(request.getSession().getAttribute(cookieName))) {
70.                         foundUser = true;
71.                     }
72.                 }
73.             }
```

Issue ID	274616
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00042.java
Line	93

Source Code

```
88.                     .println(
89.                             user
90.                                     + " has been remembered with cookie: "
91.                                     + rememberMe.getName()
92.                                     + " whose value is: "
93.                                     + rememberMe.getValue()
94.                                     + "<br/>");
95.         }
96.     } catch (java.security.NoSuchAlgorithmException e) {
97.         System.out.println("Problem executing SecureRandom.nextInt() - TestCase");
98.         throw new ServletException(e);
```

Issue ID	274626
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00061.java

Line	59

Source Code

```
54.
55.     String param = "noCookieValueSupplied";
56.     if (theCookies != null) {
57.         for (javax.servlet.http.Cookie theCookie : theCookies) {
58.             if (theCookie.getName().equals("BenchmarkTest00061")) {
59.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                 break;
61.             }
62.         }
63.     }
64.
```

Issue ID	274631
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00062.java
Line	59

Source Code

```
54.
55.     String param = "noCookieValueSupplied";
56.     if (theCookies != null) {
57.         for (javax.servlet.http.Cookie theCookie : theCookies) {
58.             if (theCookie.getName().equals("BenchmarkTest00062")) {
59.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                 break;
61.             }
62.         }
63.     }
64.
```

Issue ID	274639

File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00064.java
Line	59

Source Code

```
54.
55.     String param = "noCookieValueSupplied";
56.     if (theCookies != null) {
57.         for (javax.servlet.http.Cookie theCookie : theCookies) {
58.             if (theCookie.getName().equals("BenchmarkTest00064")) {
59.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                 break;
61.             }
62.         }
63.     }
64.
```

Issue ID	274641
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00065.java
Line	59

Source Code

```
54.
55.     String param = "noCookieValueSupplied";
56.     if (theCookies != null) {
57.         for (javax.servlet.http.Cookie theCookie : theCookies) {
58.             if (theCookie.getName().equals("BenchmarkTest00065")) {
59.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                 break;
61.             }
62.         }
63.     }
64.
```

Issue ID	274655
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00067.java
Line	59

Source Code

```
54.
55.        String param = "noCookieValueSupplied";
56.        if (theCookies != null) {
57.            for (javax.servlet.http.Cookie theCookie : theCookies) {
58.                if (theCookie.getName().equals("BenchmarkTest00067")) {
59.                    param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                    break;
61.                }
62.            }
63.        }
64.
```

Issue ID	274652
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00067.java
Line	106

Source Code

```
101.        javax.servlet.http.Cookie[] cookies = request.getCookies();
102.        if (cookies != null) {
103.            for (int i = 0; !foundUser && i < cookies.length; i++) {
104.                javax.servlet.http.Cookie cookie = cookies[i];
105.                if (cookieName.equals(cookie.getName())) {
106.                    if (cookie.getValue().equals(request.getSession().getAttribute
(cookieName))) {
107.                        foundUser = true;
108.                    }
109.                }
110.            }
111.        }
```

Issue ID	274656
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00067.java
Line	132

Source Code

```
127.                    .println(
128.                            user
129.                                    + " has been remembered with cookie: "
130.                                    + rememberMe.getName()
131.                                    + " whose value is: "
132.                                    + rememberMe.getValue()
133.                                    + "<br/>");
134.        }
135.        response.getWriter().println("Weak Randomness Test java.lang.Math.random() executed");
136.    }
137. }
```

Issue ID	274658
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00070.java
Line	59

Source Code

```
54.
55.        String param = "noCookieValueSupplied";
56.        if (theCookies != null) {
57.            for (javax.servlet.http.Cookie theCookie : theCookies) {
58.                if (theCookie.getName().equals("BenchmarkTest00070")) {
```

```
                    param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
                    break;
                }
            }
        }

```

Issue ID	274664
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00076.java
Line	59

Source Code

```

        String param = "noCookieValueSupplied";
        if (theCookies != null) {
            for (javax.servlet.http.Cookie theCookie : theCookies) {
                if (theCookie.getName().equals("BenchmarkTest00076")) {
                    param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
                    break;
                }
            }
        }

```

Issue ID	274668
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00077.java
Line	59

Source Code

```

        String param = "noCookieValueSupplied";
```

```
56.     if (theCookies != null) {
57.         for (javax.servlet.http.Cookie theCookie : theCookies) {
58.             if (theCookie.getName().equals("BenchmarkTest00077")) {
59.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                 break;
61.             }
62.         }
63.     }
64.
```

Issue ID	274678
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00084.java
Line	59

Source Code

```
54.
55.     String param = "noCookieValueSupplied";
56.     if (theCookies != null) {
57.         for (javax.servlet.http.Cookie theCookie : theCookies) {
58.             if (theCookie.getName().equals("BenchmarkTest00084")) {
59.                 param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                 break;
61.             }
62.         }
63.     }
64.
```

Issue ID	274675
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00084.java
Line	87

Source Code

```
82.        javax.servlet.http.Cookie[] cookies = request.getCookies();
83.        if (cookies != null) {
84.            for (int i = 0; !foundUser && i < cookies.length; i++) {
85.                javax.servlet.http.Cookie cookie = cookies[i];
86.                if (cookieName.equals(cookie.getName())) {
87.                    if (cookie.getValue().equals(request.getSession().getAttribute(cookieName))) {
88.                        foundUser = true;
89.                    }
90.                }
91.            }
92.        }
```

Issue ID 274676

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00084.java

Line 111

Source Code

```
106.                    .println(
107.                            user
108.                                    + " has been remembered with cookie: "
109.                                    + rememberMe.getName()
110.                                    + " whose value is: "
111.                                    + rememberMe.getValue()
112.                                    + "<br/>");
113.        }
114.
115.        response.getWriter().println("Weak Randomness Test java.util.Random.nextInt() executed");
116.    }
```

Issue ID 274684

BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode

File	/BenchmarkTest00086.java
Line	59

Source Code

```
54.
55.         String param = "noCookieValueSupplied";
56.         if (theCookies != null) {
57.             for (javax.servlet.http.Cookie theCookie : theCookies) {
58.                 if (theCookie.getName().equals("BenchmarkTest00086")) {
59.                     param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                     break;
61.                 }
62.             }
63.         }
64.
```

Issue ID	274688
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00086.java
Line	90

Source Code

```
85.         javax.servlet.http.Cookie[] cookies = request.getCookies();
86.         if (cookies != null) {
87.             for (int i = 0; !foundUser && i < cookies.length; i++) {
88.                 javax.servlet.http.Cookie cookie = cookies[i];
89.                 if (cookieName.equals(cookie.getName())) {
90.                     if (cookie.getValue().equals(request.getSession().getAttribute
(cookieName))) {
91.                         foundUser = true;
92.                     }
93.                 }
94.             }
95.         }
```

Issue ID	274689
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00086.java
Line	115

Source Code

```
110.                     .println(
111.                             user
112.                                     + " has been remembered with cookie: "
113.                                     + rememberMe.getName()
114.                                     + " whose value is: "
115.                                     + rememberMe.getValue()
116.                                     + "<br/>");
117.         }
118.
119.         response.getWriter().println("Weak Randomness Test java.util.Random.nextLong() executed");
120.     }
```

Issue ID	274695
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00087.java
Line	59

Source Code

```
54.
55.         String param = "noCookieValueSupplied";
56.         if (theCookies != null) {
57.             for (javax.servlet.http.Cookie theCookie : theCookies) {
58.                 if (theCookie.getName().equals("BenchmarkTest00087")) {
59.                     param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                     break;
61.                 }
62.             }
63.         }
64.
```

Issue ID	274698
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00092.java
Line	59

Source Code

```
54.
55.         String param = "noCookieValueSupplied";
56.         if (theCookies != null) {
57.             for (javax.servlet.http.Cookie theCookie : theCookies) {
58.                 if (theCookie.getName().equals("BenchmarkTest00092")) {
59.                     param = java.net.URLDecoder.decode(theCookie.getValue(), "UTF-8");
60.                     break;
61.                 }
62.             }
63.         }
64.
```

● [Rule Name] Exposure of system information (Medium, Java)

The System Information Exposure finds instances of displaying system information through error messages or by other means.

If error messages openly displayed by the application contain sensitive information regarding execution environments or user data, attackers can exploit the information for malicious actions. Showing exception names or stack traces can help attackers identify the program''s internal structure.

Make sure error messages are shown only to relevant users and contain minimum necessary information. Have exceptions internally handled in source code and do not allow them to trigger an error message containing sensitive information.

- CWE 660 4.14

    - 209 - Generation of Error Message Containing Sensitive Information

- 무기체계 소프트웨어 보안약점 점검 목록
    - CWE-209
    - CWE-497

- 소프트웨어 보안약점 진단가이드 2021
    - 오류메시지 정보 노출

Dangerous Example

```
1. try {
2.    ...
3. } catch (Exception e) {
4.    e.printStackTrace();
5. }
```

Line 4: Internal information is exposed by printed exception names or stack traces.

Safe Example

```
1. try {
2.    ...
3. } catch (Exception e) {
4.    logger.error("Connection Exception occurred");
5. }
```

Line 4: Do not print the exception names or stack traces.

Issue ID	274456
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DataBaseServer.java
Line	70

Source Code

```
65.             java.sql.PreparedStatement statement = connection.prepareStatement(sql);
66.             statement.execute();
67.             org.owasp.benchmark.helpers.DatabaseHelper.printResults(statement, sql, resp);
68.         } catch (java.sql.SQLException e) {
69.             if (org.owasp.benchmark.helpers.DatabaseHelper.hideSQLErrors) {
70.                 e.printStackTrace();
71.                 resp.add(new XMLMessage("Error processing request: " + e.getMessage()));
72.                 return new ResponseEntity<List<XMLMessage>>(resp, HttpStatus.OK);
73.             } else throw new ServletException(e);
74.         }
75.         return new ResponseEntity<List<XMLMessage>>(resp, HttpStatus.OK);
```

Issue ID	274465
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/DatabaseHelper.java
Line	113

Source Code

```
108.            conn.commit();
109.            initData();
110.
111.            System.out.println("DataBase tables/procedures created.");
112.        } catch (Exception e1) {
113.            System.out.println(
114.                    "Problem with database table/procedure creations: " + e1.getMessage());
115.        }
116.    }
117.
118.    public static java.sql.Statement getSqlStatement() {
```

Issue ID	274463
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java
Line	152

Source Code

```
147.
148.            executeSQLCommand(
149.                    "INSERT INTO EMPLOYEE (first_name, last_name, salary) VALUES ('foo', 'bar', 34567)");
150.            conn.commit();
151.        } catch (Exception e1) {
152.            System.out.println("Problem with database init/reset: " + e1.getMessage());
153.        }
154.    }
155.
156.    public static java.sql.Connection getSqlConnection() {
157.        if (conn == null) {
```

Issue ID	274462
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /DatabaseHelper.java
Line	165

Source Code

```
160.                DataSource datasource = (DataSource) ctx.lookup("java:comp/env/jdbc/BenchmarkDB");
161.                conn = datasource.getConnection();
162.                conn.setAutoCommit(false);
163.            } catch (SQLException | NamingException e) {
164.                System.out.println("Problem with getSqlConnection.");
165.                e.printStackTrace();
166.            }
167.        }
```

```
168.         return conn;
169.     }
170.
```

Issue ID	274467
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /LDAPManager.java
Line	52

Source Code

```
47.     public LDAPManager() {
48.         try {
49.             ctx = getDirContext();
50.         } catch (NamingException e) {
51.             // FIXME: Don't eat exceptions!
52.             System.out.println("Failed to get Directory Context: " + e.getMessage());
53.             e.printStackTrace();
54.         }
55.     }
56.
57.     protected Hashtable<Object, Object> createEnv() {
```

Issue ID	274468
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /LDAPManager.java
Line	53

Source Code

```
48.         try {
49.             ctx = getDirContext();
50.         } catch (NamingException e) {
51.             // FIXME: Don't eat exceptions!
52.             System.out.println("Failed to get Directory Context: " + e.getMessage());
53.            e.printStackTrace();
54.        }
55.    }
56.
57.    protected Hashtable<Object, Object> createEnv() {
58.        Hashtable<Object, Object> env = new Hashtable<Object, Object>();
```

Issue ID	274469
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/LDAPManager.java
Line	85

Source Code

```
80.
81.        try {
82.            iniDirContext.bind(name, ctx, matchAttrs);
83.        } catch (NamingException e) {
84.            if (!e.getMessage().contains("ENTRY_ALREADY_EXISTS")) {
85.                System.out.println("Record already exist or an error occurred: " + e.getMessage());
86.            }
87.        }
88.
89.        return true;
90.    }
```

Issue ID	274470
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/LDAPManager.java
Line	130

Source Code

```
125.            ctx.close();
126.
127.            return true;
128.        } catch (Exception e) {
129.            System.out.println("LDAP error search: ");
130.            e.printStackTrace();
131.            return false;
132.        }
133.    }
134.
135.    public DirContext getDirContext() throws NamingException {
```

Issue ID	274473
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /LDAPServer.java
Line	103

Source Code

```
98.             workDir.mkdirs();
99.             System.setProperty("workingDiretory", workDir.getPath());
100.
101.            init();
102.        } catch (Exception e) {
103.            System.out.println("Error initializing LDAP Server: " + e.getMessage());
104.            e.printStackTrace();
105.        }
106.
107.        LDAPManager emd = new LDAPManager();
108.        LDAPPerson ldapP = new LDAPPerson();
```

Issue ID	274474
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers /LDAPServer.java
Line	104

Source Code

```
99.             System.setProperty("workingDiretory", workDir.getPath());
100.
101.              init();
102.          } catch (Exception e) {
103.              System.out.println("Error initializing LDAP Server: " + e.getMessage());
104.              e.printStackTrace();
105.          }
106.
107.          LDAPManager emd = new LDAPManager();
108.          LDAPPerson ldapP = new LDAPPerson();
109.          ldapP.setName("foo");
```

Issue ID	274476
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	107

Source Code

```
102.              safeDocBuilderFactory.setFeature(
103.                      "http://apache.org/xml/features/disallow-doctype-decl", true);
104.          } catch (ParserConfigurationException e) {
105.              System.out.println(
106.                      "ERROR: couldn't set http://apache.org/xml/features/disallow-
doctype-decl");
107.              e.printStackTrace();
108.          }
109.
110.          File tempDir = new File(TESTFILES_DIR);
111.          if (!tempDir.exists()) {
112.              tempDir.mkdir();
```

Issue ID	274477

File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	119

Source Code

```
114.        try {
115.            PrintWriter out = new PrintWriter(testFile);
116.            out.write("Test is a test file.₩n");
117.            out.close();
118.        } catch (FileNotFoundException e) {
119.            e.printStackTrace();
120.        }
121.        File testFile2 = new File(TESTFILES_DIR + "SafeText");
122.        try {
123.            PrintWriter out = new PrintWriter(testFile2);
124.            out.write("Test is a 'safe' test file.₩n");
```

Issue ID	274478
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	127

Source Code

```
122.        try {
123.            PrintWriter out = new PrintWriter(testFile2);
124.            out.write("Test is a 'safe' test file.₩n");
125.            out.close();
126.        } catch (FileNotFoundException e) {
127.            e.printStackTrace();
128.        }
129.        File secreTestFile = new File(TESTFILES_DIR + "SecretFile");
130.        try {
131.            PrintWriter out = new PrintWriter(secreTestFile);
132.            out.write("Test is a 'secret' file that no one should find.₩n");
```

Issue ID	274479
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	135

Source Code

```
130.         try {
131.             PrintWriter out = new PrintWriter(secreTestFile);
132.             out.write("Test is a 'secret' file that no one should find.₩n");
133.             out.close();
134.         } catch (FileNotFoundException e) {
135.             e.printStackTrace();
136.         }
137.     }
138.
139.     // The target script is exploded out of the WAR file. When this occurs, the file
140.     // loses its execute permissions. So this hack adds the required execute permissions back.
```

Issue ID	274486
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	258

Source Code

```
253.             out.write(ESAPI.encoder().encodeForHTML(s));
254.             out.write("<br>");
255.         }
256.     } catch (IOException e) {
257.         System.out.println("An error occurred while reading OSCommandResults");
258.         e.printStackTrace();
259.     }
```

```
260.    }
261.
262.    // A method used by the Benchmark JAVA test cases to format OS Command
Output
263.    // This version is only used by the Web Services test cases.
```

Issue ID	274487
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	289

Source Code

```
284.            }
285.
286.            resp.add(new XMLMessage(outError.toString()));
287.        } catch (IOException e) {
288.            System.out.println("An error occurred while reading
OSCommandResults");
289.            e.printStackTrace();
290.        }
291.    }
292.
293.    public static File getFileFromClasspath(String fileName, ClassLoader
classLoader) {
294.        URL url = classLoader.getResource(fileName);
```

Issue ID	274481
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	301

Source Code

```
296.            try {
297.                return new File(url.toURI().getPath());
298.            } catch (URISyntaxException e) {
299.                System.out.println(
300.                        "The file '" + fileName + "' cannot be loaded from the classpath.");
301.                e.printStackTrace();
302.            }
303.        } else System.out.println("The file '" + fileName + "' cannot be found on the
classpath.");
304.        return null;
305.    }
306.
```

Issue ID	274482
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	313

Source Code

```
308.        if (!file.exists()) {
309.            try {
310.                System.out.println("Can't find file to get lines from: " + file.
getCanonicalFile());
311.            } catch (IOException e) {
312.                System.out.println("Can't find file to get lines from.");
313.                e.printStackTrace();
314.            }
315.            return null;
316.        }
317.
318.        List<String> sourceLines = new ArrayList<String>();
```

Issue ID	274483
	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.

File	java
Line	331

Source Code

```
326.        } catch (Exception e) {
327.            try {
328.                System.out.println("Problem reading contents of file: " + file.getCanonicalFile());
329.            } catch (IOException e2) {
330.                System.out.println("Problem reading file to get lines from.");
331.                e2.printStackTrace();
332.            }
333.            e.printStackTrace();
334.        }
335.
336.        return sourceLines;
```

Issue ID	274484
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	333

Source Code

```
328.                System.out.println("Problem reading contents of file: " + file.getCanonicalFile());
329.            } catch (IOException e2) {
330.                System.out.println("Problem reading file to get lines from.");
331.                e2.printStackTrace();
332.            }
333.            e.printStackTrace();
334.        }
335.
336.        return sourceLines;
337.    }
338.
```

Issue ID	274488
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	388

Source Code

```
383.            FileOutputStream fos = new FileOutputStream(f, true);
384.            os = new PrintStream(fos);
385.            os.println(line);
386.        } catch (IOException e1) {
387.            result = false;
388.            e1.printStackTrace();
389.        } finally {
390.            os.close();
391.        }
392.
393.        return result;
```

Issue ID	274480
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/helpers/Utils.java
Line	417

Source Code

```
412.                cipher.init(javax.crypto.Cipher.ENCRYPT_MODE, publicKey);
413.            } catch (NoSuchAlgorithmException
414.                    | NoSuchProviderException
415.                    | NoSuchPaddingException
416.                    | InvalidKeyException e) {
417.                e.printStackTrace();
418.            }
419.        }
420.        return cipher;
```

```
421.    }
422.
```

Issue ID	274562
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	75

Source Code

```
70.            fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
71.            fis = new java.io.FileInputStream(new java.io.File(fileName));
72.            byte[] b = new byte[1000];
73.            int size = fis.read(b);
74.            response.getWriter()
75.                    .println(
76.                            "The beginning of file: '"
77.                                    + org.owasp.esapi.ESAPI.encoder().encodeForHTML
(fileName)
78.                                    + "' is:₩n₩n"
79.                                    + org.owasp
80.                                            .esapi
```

Issue ID	274561
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00001.java
Line	85

Source Code

```
80.                                            .esapi
81.                                            .ESAPI
82.                                            .encoder()
83.                                            .encodeForHTML(new String(b, 0, size)));
84.        } catch (Exception e) {
```

```
85.         System.out.println("Couldn't open FileInputStream on file: '" + fileName +
"'");
86.         response.getWriter()
87.                 .println(
88.                         "Problem getting FileInputStream: "
89.                                 + org.owasp
90.                                         .esapi
```

Issue ID	274572
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00002.java
Line	74

Source Code

```
69.     try {
70.         fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + param;
71.
72.         fos = new java.io.FileOutputStream(fileName, false);
73.         response.getWriter()
74.                 .println(
75.                         "Now ready to write to file: "
76.                                 + org.owasp.esapi.ESAPI.encoder().encodeForHTML
(fileName));
77.
78.     } catch (Exception e) {
79.         System.out.println("Couldn't open FileOutputStream on file: '" + fileName
+ "'");
```

Issue ID	274574
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00003.java
Line	101

Source Code

```
96.             "hash_value="
97.                 + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
98.                 + "₩n");
99.         fw.close();
100.         response.getWriter()
101.             .println(
102.                 "Sensitive value '"
103.                     + org.owasp
104.                         .esapi
105.                         .ESAPI
106.                         .encoder()
```

Issue ID 274495

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00005.java

Line 118

Source Code

```
113.           | java.security.InvalidKeyException
114.           | java.security.InvalidAlgorithmParameterException e) {
115.         response.getWriter()
116.             .println(
117.                 "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
118.         e.printStackTrace(response.getWriter());
119.         throw new ServletException(e);
120.       }
121.     }
122. }
```

Issue ID 274502

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00020.java

Line	108

Source Code

```
103.
104.        } catch (java.security.NoSuchAlgorithmException e) {
105.            response.getWriter()
106.                    .println(
107.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
108.            e.printStackTrace(response.getWriter());
109.            throw new ServletException(e);
110.        } catch (java.security.NoSuchProviderException e) {
111.            response.getWriter()
112.                    .println(
113.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274503
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00020.java
Line	114

Source Code

```
109.            throw new ServletException(e);
110.        } catch (java.security.NoSuchProviderException e) {
111.            response.getWriter()
112.                    .println(
113.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
114.            e.printStackTrace(response.getWriter());
115.            throw new ServletException(e);
116.        } catch (javax.crypto.NoSuchPaddingException e) {
117.            response.getWriter()
118.                    .println(
119.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274504
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00020.java
Line	120

Source Code

```
115.            throw new ServletException(e);
116.        } catch (javax.crypto.NoSuchPaddingException e) {
117.            response.getWriter()
118.                    .println(
119.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
120.            e.printStackTrace(response.getWriter());
121.            throw new ServletException(e);
122.        } catch (javax.crypto.IllegalBlockSizeException e) {
123.            response.getWriter()
124.                    .println(
125.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274505
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00020.java
Line	126

Source Code

```
121.            throw new ServletException(e);
122.        } catch (javax.crypto.IllegalBlockSizeException e) {
123.            response.getWriter()
124.                    .println(
125.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
```

```
126.            e.printStackTrace(response.getWriter());
127.            throw new ServletException(e);
128.        } catch (javax.crypto.BadPaddingException e) {
129.            response.getWriter()
130.                    .println(
131.                            "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274506
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00020.java
Line	132

Source Code

```
127.            throw new ServletException(e);
128.        } catch (javax.crypto.BadPaddingException e) {
129.            response.getWriter()
130.                    .println(
131.                            "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
132.            e.printStackTrace(response.getWriter());
133.            throw new ServletException(e);
134.        } catch (java.security.InvalidKeyException e) {
135.            response.getWriter()
136.                    .println(
137.                            "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274507
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00020.java
Line	144

Source Code

```
139.            throw new ServletException(e);
140.        } catch (java.security.InvalidAlgorithmParameterException e) {
141.            response.getWriter()
142.                    .println(
143.                            "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
144.            e.printStackTrace(response.getWriter());
145.            throw new ServletException(e);
146.        }
147.        response.getWriter()
148.                .println(
149.                        "Crypto Test javax.crypto.Cipher.getInstance(java.lang.String,java.
lang.String) executed");
```

Issue ID	274603
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00023.java
Line	85

Source Code

```
80.         rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST
this servlet
81.         // e.g., /benchmark/sql-01/BenchmarkTest01001
82.         request.getSession().setAttribute(cookieName, rememberMeKey);
83.         response.addCookie(rememberMe);
84.         response.getWriter()
85.                 .println(
86.                         user
87.                                 + " has been remembered with cookie: "
88.                                 + rememberMe.getName()
89.                                 + " whose value is: "
90.                                 + rememberMe.getValue()
```

Issue ID	274614

File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00042.java
Line	88

Source Code

```
83.         rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
84.         // e.g., /benchmark/sql-01/BenchmarkTest01001
85.         request.getSession().setAttribute(cookieName, rememberMeKey);
86.         response.addCookie(rememberMe);
87.         response.getWriter()
88.                 .println(
89.                         user
90.                                 + " has been remembered with cookie: "
91.                                 + rememberMe.getName()
92.                                 + " whose value is: "
93.                                 + rememberMe.getValue()
```

Issue ID	274516
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00050.java
Line	109

Source Code

```
104.
105.     } catch (java.security.NoSuchAlgorithmException e) {
106.         response.getWriter()
107.                 .println(
108.                         "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
109.         e.printStackTrace(response.getWriter());
110.         throw new ServletException(e);
111.     } catch (java.security.NoSuchProviderException e) {
112.         response.getWriter()
113.                 .println(
114.                    "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274517
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00050.java
Line	115

Source Code

```
110.            throw new ServletException(e);
111.        } catch (java.security.NoSuchProviderException e) {
112.            response.getWriter()
113.                    .println(
114.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
115.            e.printStackTrace(response.getWriter());
116.            throw new ServletException(e);
117.        } catch (javax.crypto.NoSuchPaddingException e) {
118.            response.getWriter()
119.                    .println(
120.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274518
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00050.java
Line	121

Source Code

```
116.            throw new ServletException(e);
117.        } catch (javax.crypto.NoSuchPaddingException e) {
118.            response.getWriter()
119.                    .println(
120.                    "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
121.            e.printStackTrace(response.getWriter());
122.            throw new ServletException(e);
123.        } catch (javax.crypto.IllegalBlockSizeException e) {
124.            response.getWriter()
125.                .println(
126.                    "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274519
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00050.java
Line	127

Source Code

```
122.            throw new ServletException(e);
123.        } catch (javax.crypto.IllegalBlockSizeException e) {
124.            response.getWriter()
125.                .println(
126.                    "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
127.            e.printStackTrace(response.getWriter());
128.            throw new ServletException(e);
129.        } catch (javax.crypto.BadPaddingException e) {
130.            response.getWriter()
131.                .println(
132.                    "Problem executing crypto - javax.crypto.Cipher.getInstance
(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274520
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00050.java
Line	133

Source Code

```
128.            throw new ServletException(e);
129.        } catch (javax.crypto.BadPaddingException e) {
130.            response.getWriter()
131.                    .println(
132.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
133.            e.printStackTrace(response.getWriter());
134.            throw new ServletException(e);
135.        } catch (java.security.InvalidKeyException e) {
136.            response.getWriter()
137.                    .println(
138.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
```

Issue ID	274521
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00050.java
Line	145

Source Code

```
140.            throw new ServletException(e);
141.        } catch (java.security.InvalidAlgorithmParameterException e) {
142.            response.getWriter()
143.                    .println(
144.                            "Problem executing crypto - javax.crypto.Cipher.getInstance(java.lang.String,java.security.Provider) Test Case");
145.            e.printStackTrace(response.getWriter());
146.            throw new ServletException(e);
147.        }
148.        response.getWriter()
149.                .println(
150.                        "Crypto Test javax.crypto.Cipher.getInstance(java.lang.String,java.lang.String) executed");
```

Issue ID	274629
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00061.java
Line	76

Source Code

```
71.                           param.getBytes())));
72.         }
73.
74.         java.io.File fileTarget = new java.io.File(bar, "/Test.txt");
75.         response.getWriter()
76.             .println(
77.                 "Access to file: '"
78.                     + org.owasp
79.                         .esapi
80.                         .ESAPI
81.                         .encoder()
```

Issue ID	274632
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00062.java
Line	81

Source Code

```
76.           fileName = org.owasp.benchmark.helpers.Utils.TESTFILES_DIR + bar;
77.           fis = new java.io.FileInputStream(new java.io.File(fileName));
78.           byte[] b = new byte[1000];
79.           int size = fis.read(b);
80.           response.getWriter()
81.               .println(
82.                   "The beginning of file: '"
83.                       + org.owasp.esapi.ESAPI.encoder().encodeForHTML
(fileName)
84.                       + "' is:₩n₩n"
```

```
85.                                     + org.owasp
86.                                             .esapi
```

Issue ID 274637

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00062.java

Line 91

Source Code

```
86.                                             .esapi
87.                                             .ESAPI
88.                                             .encoder()
89.                                             .encodeForHTML(new String(b, 0, size)));
90.         } catch (Exception e) {
91.             System.out.println("Couldn't open FileInputStream on file: '" + fileName +
"'");
92.             response.getWriter()
93.                     .println(
94.                             "Problem getting FileInputStream: "
95.                                     + org.owasp
96.                                             .esapi
```

Issue ID 274646

File BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00065.java

Line 83

Source Code

```
78.             java.nio.file.Path path = java.nio.file.Paths.get(fileName);
79.             is = java.nio.file.Files.newInputStream(path, java.nio.file.
StandardOpenOption.READ);
80.             byte[] b = new byte[1000];
81.             int size = is.read(b);
```

```
82.         response.getWriter()
83.             .println(
84.                 "The beginning of file: '"
85.                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML(fileName)
86.                     + "' is:₩n₩n");
87.         response.getWriter()
88.             .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(new String(b, 0, size)));
```

Issue ID	274643
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00065.java
Line	88

Source Code

```
83.             .println(
84.                 "The beginning of file: '"
85.                     + org.owasp.esapi.ESAPI.encoder().encodeForHTML(fileName)
86.                     + "' is:₩n₩n");
87.         response.getWriter()
88.             .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(new String(b, 0, size)));
89.         is.close();
90.     } catch (Exception e) {
91.         System.out.println("Couldn't open InputStream on file: '" + fileName + "'");
92.         response.getWriter()
93.             .println(
```

Issue ID	274642
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00065.java
Line	91

Source Code

```
86.                          + "' is:₩n₩n");
87.             response.getWriter()
88.                     .println(org.owasp.esapi.ESAPI.encoder().encodeForHTML(new String(b, 0, size)));
89.             is.close();
90.         } catch (Exception e) {
91.             System.out.println("Couldn't open InputStream on file: '" + fileName + "'");
92.             response.getWriter()
93.                     .println(
94.                             "Problem getting InputStream: "
95.                                     + org.owasp
96.                                             .esapi
```

Issue ID	274648
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00067.java
Line	127

Source Code

```
122.            rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
123.            // e.g., /benchmark/sql-01/BenchmarkTest01001
124.            request.getSession().setAttribute(cookieName, rememberMeKey);
125.            response.addCookie(rememberMe);
126.            response.getWriter()
127.                    .println(
128.                            user
129.                                    + " has been remembered with cookie: "
130.                                    + rememberMe.getName()
131.                                    + " whose value is: "
132.                                    + rememberMe.getValue()
```

Issue ID	274660
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00070.java
Line	103

Source Code

```
98.             "hash_value="
99.                 + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result, true)
100.                + "₩n");
101.        fw.close();
102.        response.getWriter()
103.            .println(
104.                "Sensitive value '"
105.                    + org.owasp
106.                        .esapi
107.                        .ESAPI
108.                        .encoder()
```

Issue ID	274665
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00076.java
Line	121

Source Code

```
116.            "hash_value="
117.                + org.owasp.esapi.ESAPI.encoder().encodeForBase64(result,
true)
118.                + "₩n");
119.        fw.close();
120.        response.getWriter()
121.            .println(
122.                "Sensitive value '"
123.                    + org.owasp
```

```
124.                    .esapi
125.                    .ESAPI
126.                    .encoder()
```

Issue ID	274677
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00084.java
Line	106

Source Code

```
101.        rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
102.        // e.g., /benchmark/sql-01/BenchmarkTest01001
103.        request.getSession().setAttribute(cookieName, rememberMeKey);
104.        response.addCookie(rememberMe);
105.        response.getWriter()
106.                .println(
107.                        user
108.                                + " has been remembered with cookie: "
109.                                + rememberMe.getName()
110.                                + " whose value is: "
111.                                + rememberMe.getValue()
```

Issue ID	274683
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode/BenchmarkTest00086.java
Line	110

Source Code

```
105.        rememberMe.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
106.        // e.g., /benchmark/sql-01/BenchmarkTest01001
107.        request.getSession().setAttribute(cookieName, rememberMeKey);
```

```
108.            response.addCookie(rememberMe);
109.            response.getWriter()
110.                    .println(
111.                            user
112.                                    + " has been remembered with cookie: "
113.                                    + rememberMe.getName()
114.                                    + " whose value is: "
115.                                    + rememberMe.getValue()
```

Issue ID	274692
File	BenchmarkJava-master/src/main/java/org/owasp/benchmark/testcode /BenchmarkTest00087.java
Line	98

Source Code

```
93.        cookie.setPath(request.getRequestURI()); // i.e., set path to JUST this servlet
94.        // e.g., /benchmark/sql-01/BenchmarkTest01001
95.        response.addCookie(cookie);
96.
97.        response.getWriter()
98.                .println(
99.                        "Created cookie: 'SomeCookie': with value: '"
100.                                + org.owasp.esapi.ESAPI.encoder().encodeForHTML(str)
101.                                + "' and secure flag set to: false");
102.    }
103. }
```

● [Rule Name] Use of Components Licensed Under the Apache-2.0 (Low, Common)

The Apache License 2.0 (SPDX ID: Apache-2.0) carries an obligation for notice. You should read the full license and be aware of its use.

Open Source License

License Name Apache License 2.0

Open Source Component

Issue ID	274447
File	BenchmarkJava-master/pom.xml
Component Name	commons-lang:commons-lang
Component Version	2.6

Open Source License

License Name	Apache License 2.0

Open Source Component

Issue ID	274452
File	BenchmarkJava-master/pom.xml
Component Name	commons-dbcp:commons-dbcp
Component Version	1.4

● [Rule Name] Use of Components Licensed Under the CDDL-1.0 (Medium, Common)

The Common Development and Distribution License 1.0 (SPDX ID: CDDL-1.0) has obligations for notice, obligations for source code disclosure based on file. You should read the full license to be aware of its use.

Open Source License

License Name	Common Development and Distribution License 1.0

Open Source Component

Issue ID	274435
File	BenchmarkJava-master/pom.xml

Component Name	javax:javaee-api
Component Version	8.0.1

● [Rule Name] Use of Components Licensed Under the CDDL-1.1 (Medium, Common)

The Common Development and Distribution License 1.1 (SPDX ID: CDDL-1.1) has obligations for notice, obligations for source code disclosure based on file. You should read the full license to be aware of its use.

Open Source License

License Name	Common Development and Distribution License 1.1

Open Source Component

Issue ID	274436
File	BenchmarkJava-master/pom.xml
Component Name	javax:javaee-api
Component Version	8.0.1

Open Source License

License Name	Common Development and Distribution License 1.1

Open Source Component

Issue ID	274455
File	BenchmarkJava-master/pom.xml
Component Name	com.sun.jersey:jersey-servlet
Component Version	1.19.4

● [Rule Name] Use of Components Licensed Under the GPL-2.0-with-classpath-exception (High, Common)

The GNU General Public License v2.0 w/Classpath exception (SPDX ID: GPL-2.0-with-classpath-exception) has no obligations for notice, no limitations, and no source code disclosure scope information. You should check the full text of the license.

Open Source License

License Name	GNU General Public License v2.0 w/Classpath exception

Open Source Component

Issue ID	274454
File	BenchmarkJava-master/pom.xml
Component Name	com.sun.jersey:jersey-servlet
Component Version	1.19.4

● [Rule Name] Use of Components Licensed Under the W3C (Low, Common)

The W3C Software Notice and License (2002-12-31) (SPDX ID: W3C) has obligations for notice. You should check the full text of the license to be aware of its use.

Open Source License

License Name	W3C Software Notice and License (2002-12-31)

Open Source Component

Issue ID	274453
File	BenchmarkJava-master/pom.xml
Component Name	xml-apis:xml-apis
Component Version	1.4.01

● [Rule Name] Use of Vulnerable Component (Critical, Common)

The One of the most important steps you can take to ensure the security of your application is to avoid using components with known security vulnerabilities.

Vulnerable components must be updated to the latest secure version or replaced with other secure components.

Known Vulnerability in Component

Vulnerability Name	CVE-2024-30171
Vulnerability Description	An issue was discovered in Bouncy Castle Java TLS API and JSSE Provider before 1.78. Timing-based leakage may occur in RSA based handshakes because of exception processing.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	-	-	-	-	-	-	-	-

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274437
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	GHSA-wjxj-5m7g-mg7q
Vulnerability Description	Bouncy Castle for Java before 1.73 contains a potential Denial of Service (DoS) issue within the Bouncy Castle org.bouncycastle.openssl.PEMParser class. This class parses OpenSSL PEM encoded streams containing X.509 certificates, PKCS8 encoded keys, and PKCS7 objects. Parsing a file that has crafted ASN.1 data through the PEMParser causes an OutOfMemoryError, which can enable a denial of service attack.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	LOCAL	LOW	NONE	REQUIRED	UNCHANGED	NONE	NONE	HIGH

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274438
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	CVE-2023-33202
	Bouncy Castle for Java before 1.73 contains a potential Denial of Service (DoS) issue within the Bouncy Castle org.bouncycastle.openssl.PEMParser class. This class parses OpenSSL PEM encoded streams containing X.509

Vulnerability Description	certificates, PKCS8 encoded keys, and PKCS7 objects. Parsing a file that has crafted ASN.1 data through the PEMParser causes an OutOfMemoryError, which can enable a denial of service attack. (For users of the FIPS Java API: BC-FJA 1.0.2.3 and earlier are affected; BC-FJA 1.0.2.4 is fixed.)

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	LOCAL	LOW	NONE	REQUIRED	UNCHANGED	NONE	NONE	HIGH

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274439
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	GHSA-8xfc-gm6g-vgpv
Vulnerability Description	An issue was discovered in ECCurve.java and ECCurve.cs in Bouncy Castle Java (BC Java) before 1.78, BC Java LTS before 2.73.6, BC-FJA before 1.0.2.5, and BC C# .Net before 2.3.1. Importing an EC certificate with crafted F2m parameters can lead to excessive CPU consumption during the evaluation of the curve parameters.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	LOW	NONE	NONE	UNCHANGED	NONE	NONE	LOW

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274440
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	CVE-2023-33201
Vulnerability Description	Bouncy Castle For Java before 1.74 is affected by an LDAP injection vulnerability. The vulnerability only affects applications that use an LDAP CertStore from Bouncy Castle to validate X.509 certificates. During the certificate validation process, Bouncy Castle inserts the certificate's Subject Name into an LDAP search filter without any escaping, which leads to an LDAP injection vulnerability.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	LOW	NONE	NONE	UNCHANGED	LOW	NONE	NONE

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274441
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

Update the component version to between 1.46 to 1.48.

Known Vulnerability in Component

Vulnerability Name	CVE-2024-29857
Vulnerability Description	An issue was discovered in ECCurve.java and ECCurve.cs in Bouncy Castle Java (BC Java) before 1.78, BC Java LTS before 2.73.6, BC-FJA before 1.0.2.5, and BC C# .Net before 2.3.1. Importing an EC certificate with crafted F2m parameters can lead to excessive CPU consumption during the evaluation of the curve parameters.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	-	-	-	-	-	-	-	-

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274442
File	BenchmarkJava-master/pom.xml

Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	GHSA-m44j-cfrm-g8qc
Vulnerability Description	An issue was discovered in Bouncy Castle Java Cryptography APIs starting in 1.73 and before 1.78. An Ed25519 verification code infinite loop can occur via a crafted signature and public key.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	LOW	NONE	NONE	UNCHANGED	NONE	NONE	LOW

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274443
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	CVE-2024-30172
Vulnerability Description	An issue was discovered in Bouncy Castle Java Cryptography APIs before 1.78. An Ed25519 verification code infinite loop can occur via a crafted signature and public key.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	-	-	-	-	-	-	-	-

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274444
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	GHSA-hr8g-6v94-x4m9
	Bouncy Castle provides the `X509LDAPCertStoreSpi.java` class which can be used in conjunction with the CertPath API for validating certificate paths. Pre-1.73 the implementation did not check the X.500 name of any certificate, subject, or issuer being passed in for LDAP wild cards, meaning the presence of a wild car may lead to

Vulnerability Description	Information Disclosure. A potential attack would be to generate a self-signed certificate with a subject name that contains special characters, e.g: `CN=Subject*)(objectclass=`. This will be included into the filter and provides the attacker ability to specify additional attributes in the search query. This can be exploited as a blind LDAP injection: an attacker can enumerate valid attribute values using the boolean blind injection technique. The exploitation depends on the structure of the target LDAP directory, as well as what kind of errors are exposed to the user. Changes to the `X509LDAPCertStoreSpi.java` class add the additional checking of any X.500 name used to correctly escape wild card characters.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	LOW	NONE	NONE	UNCHANGED	LOW	NONE	NONE

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274445
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

Update the component version to between 1.46 to 1.48.

Known Vulnerability in Component

Vulnerability Name	GHSA-v435-xc8x-wvr9

Vulnerability Description	An issue was discovered in Bouncy Castle Java TLS API and JSSE Provider before 1.78. Timing-based leakage may occur in RSA based handshakes because of exception processing.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	HIGH	NONE	NONE	UNCHANGED	HIGH	NONE	NONE

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274446
File	BenchmarkJava-master/pom.xml
Component Name	org.bouncycastle:bcprov-jdk15on
Component Version	1.70

Remedial Recommendation

No recommendation to remediate the vulnerability.

Known Vulnerability in Component

Vulnerability Name	CVE-2020-25638
Vulnerability Description	A flaw was found in hibernate-core in versions prior to and including 5.4.23.Final. A SQL injection in the implementation of the JPA Criteria API can permit unsanitized literals when a literal is used in the SQL comments of the query. This flaw could allow an attacker to access unauthorized information or possibly conduct further attacks. The highest threat from this vulnerability is to data confidentiality and integrity.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	HIGH	NONE	NONE	UNCHANGED	HIGH	HIGH	NONE

CVSS 2

Score	AV	AC	Au	C	I	A
-	NETWORK	MEDIUM	NONE	PARTIAL	PARTIAL	NONE

Open Source Component

Issue ID	274448
File	BenchmarkJava-master/pom.xml
Component Name	org.hibernate:hibernate-core
Component Version	3.6.10.Final

Remedial Recommendation

Update the component version to between 5.4.24.Final to 5.6.15.Final.

Known Vulnerability in Component

Vulnerability Name	GHSA-j8jw-g6fq-mp7h
Vulnerability Description	A flaw was found in hibernate-core in versions prior to 5.3.20.Final and in 5.4.0.Final up to and including 5.4.23. Final. A SQL injection in the implementation of the JPA Criteria API can permit unsanitized literals when a literal is used in the SQL comments of the query. This flaw could allow an attacker to access unauthorized information or possibly conduct further attacks. The highest threat from this vulnerability is to data confidentiality and integrity.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	HIGH	NONE	NONE	UNCHANGED	HIGH	HIGH	NONE

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274449
File	BenchmarkJava-master/pom.xml
Component Name	org.hibernate:hibernate-core
Component Version	3.6.10.Final

Remedial Recommendation

Update the component version to between 5.4.24.Final to 5.6.15.Final.

Known Vulnerability in Component

Vulnerability Name	GHSA-8grg-q944-cch5
Vulnerability Description	A flaw was found in Hibernate ORM in versions before 5.3.18, 5.4.18 and 5.5.0.Beta1. A SQL injection in the implementation of the JPA Criteria API can permit unsanitized literals when a literal is used in the SELECT or GROUP BY parts of the query. This flaw could allow an attacker to access unauthorized information or possibly conduct further attacks.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	LOW	LOW	NONE	UNCHANGED	HIGH	NONE	NONE

CVSS 2

Score	AV	AC	Au	C	I	A
-	-	-	-	-	-	-

Open Source Component

Issue ID	274450

File	BenchmarkJava-master/pom.xml
Component Name	org.hibernate:hibernate-core
Component Version	3.6.10.Final

Remedial Recommendation

Update the component version to between 5.5.0.Beta1 to 5.6.15.Final.

Known Vulnerability in Component

Vulnerability Name	CVE-2019-14900
Vulnerability Description	A flaw was found in Hibernate ORM in versions before 5.3.18, 5.4.18 and 5.5.0.Beta1. A SQL injection in the implementation of the JPA Criteria API can permit unsanitized literals when a literal is used in the SELECT or GROUP BY parts of the query. This flaw could allow an attacker to access unauthorized information or possibly conduct further attacks.

CVSS 3

Score	AV	AC	PR	UI	S	C	I	A
-	NETWORK	LOW	LOW	NONE	UNCHANGED	HIGH	NONE	NONE

CVSS 2

Score	AV	AC	Au	C	I	A
-	NETWORK	LOW	SINGLE	PARTIAL	NONE	NONE

Open Source Component

Issue ID	274451
File	BenchmarkJava-master/pom.xml
Component Name	org.hibernate:hibernate-core
Component Version	3.6.10.Final

Remedial Recommendation

Update the component version to between 5.5.0.Beta1 to 5.6.15.Final.

■ Excluded Issues

No issue has been excluded.

Detection Rules

Type	Risk Level	Language	Name
Code	Low	ABAP	Too long line
Code	Low	ABAP	Improper WHEN OTHERS Position
Code	Low	ABAP	Violation of naming rule for DATA variable
Code	Low	ABAP	Violation of naming rule for class
Code	Low	ABAP	Violation of naming rule for form
Code	Low	ABAP	Violation of naming rule for function
Code	Low	ABAP	Violation of naming rule for interface
Code	Low	ABAP	Violation of naming rule for method
Code	Low	ABAP	Violation of naming rule for REPORT
Code	Low	ABAP	Violation of naming rule for program
Code	Low	C	Assignment of Integer-type Null to Pointer Variable
Code	Trivial	C	Exposal of Internal Implementation of Struct/Union
Code	Low	C++	Non-standard escape sequence
Code	Low	C++	Duplicated name in different scope
Code	Low	C++	Non-unique user defined type name
Code	Low	C++	Duplicated tag name
Code	Low	C++	Reused static identifier
Code	Trivial	C++	Reused identifier
Code	Trivial	C++	Missing size specification in type definition
Code	Low	C++	Improper operand of logical operation
Code	Low	C++	Use of continue statement
Code	Trivial	C++	Missing default case in switch statement
Code	Low	C++	Too high pointer level
Code	Low	C++	Too complex function
Code	Low	C++	Non-boolean value as operand of logical operation
Code	Low	C++	Multiple break statements
Code	Trivial	C++	Case statement in nested block
Code	Low	C++	Missing identifier for parameter
Code	Trivial	C++	Mismatched name of argument
Code	Low	C++	Following include statement
Code	Low	C++	Inconsistent parameter declaration
Code	Trivial	C++	Non-const member function

Code	Trivial	C++	Declaration of external global variable in header file
Code	Trivial	C++	Missing definition of inline function in header file
Code	Trivial	C++	Template definition in source file
Code	Low	C++	Non-unique Identifier
Code	Trivial	C++	Use of ambiguous grammar
Code	Low	C++	Confusing character in identifier
Code	Low	C++	Multiple type name in single type definition
Code	Low	C++	User defined array type
Code	Low	C++	Use of C-style Array
Code	Trivial	C++	Violation of rule for * and & tokens
Code	Trivial	C++	Use of unsigned type
Code	Low	C++	Improper comparison with constant value
Code	Trivial	C++	Use of Ternary Operator
Code	Low	C++	Non-static overloaded new operator
Code	Low	C++	Violation of rule for white space in preprocess statement
Code	Low	C++	Violation of rule for white space in nested preprocess statements
Code	Low	C++	Use of #if directive
Code	Low	C++	Path specifier in include statement
Code	Low	C++	Use of Null Macro
Code	Low	C++	Violation of rule of class organization
Code	Low	C++	Illegible identifier
Code	Low	C++	Type declaration in source file
Code	Low	C++	Non-member function at outside of anonymous namespace
Code	Low	C++	Member function in structure
Code	Low	C++	Comparing enumeration type with integer value
Code	Trivial	C++	Variable name of lower scope including variable name of upper scope
Code	Trivial	C++	Violation of naming rule for variable (BSSC)
Code	Trivial	C++	Violation of naming rule for function
Code	Trivial	C++	Violation of naming rule for user defined type
Code	Trivial	C++	Violation of naming rule for class
Code	Low	C++	Improper access specifier
Code	Low	C++	Early variable definition

Code	Low	C++	Redundant casting
Code	Low	C++	Improperly Formatted Macro Name
Code	Medium	C++	Throwing unspecified exception
Code	Medium	C++	Missing call of sizeof
Code	Medium	C++	Wrong length of field array
Code	Medium	C++	Duplicated parameters
Code	Trivial	C++	Code in Comment
Code	Trivial	C++	Duplicate Definition of Virtual Function
Code	Trivial	C++	Missing overriding of pure virtual function
Code	Medium	C++	Invalid Comparison with Loop Counter
Code	Trivial	C++	Duplicate Loop Counter
Code	Low	C++	Out-of-scope case Statement
Code	Trivial	C++	Unnamed Namespace of Header File
Code	Trivial	C++	Non-main Function Uses Name of main()
Code	Low	C++	Unused Function Return
Code	Low	C++	Overloading of Function in Namespace
Code	Low	C++	Cast of cvalue Expression
Code	Low	C++	Array Decayed to Pointer
Code	Trivial	C++	Invalid Preprocessing Token
Code	Trivial	C++	Variable and Function Declaration in Global Scope
Code	Trivial	C++	Missing Call to Superclass Constructor
Code	Trivial	C++	Throwing of Unlisted Exception
Code	Trivial	C++	Template Specialization Declaration in Another File
Code	Trivial	C++	Violation of One Definition Rule
Code	Trivial	C++	Too many function calls
Code	Medium	C++	Use of goto statement
Code	Low	C++	Too many return statements
Code	Trivial	C++	Non-macro constant
Code	Trivial	C++	Constant in condition of loop statement
Code	Trivial	C++	Variable declaration at middle of function
Code	Trivial	C++	Too many consecutive if statements
Code	Trivial	C++	Unprefixed Constant in Enum
Code	Trivial	C++	Violation of Hungarian notation
Code	High	C++	Missing initialization of variable
Code	Trivial	C++	Violation of naming rule for macro
Code	Trivial	C++	Violation of naming rule for constant

Code	Medium	C++	Braced Block Starting in Same Line as Condition
Code	Low	C++	Duplicate Name
Code	Trivial	C++	Violation of naming rule for global variable
Code	Trivial	C++	Too Long Name
Code	Medium	C++	Missing explicit array size
Code	Trivial	C++	Violation of format of block comments
Code	Trivial	C++	Too long source code before comment
Code	Trivial	C++	Missing blank line before comment
Code	Trivial	C++	Violation of indentation rule for comment
Code	Low	C++	Invalid Enumerator Operation
Code	Trivial	C++	typedef Declaration of Numeric Type without Signedness
Code	Trivial	C++	Noncompliant Macro Name Length
Code	Trivial	C++	Duplicate Name with Macro
Code	Trivial	C++	switch Statement with Starting or without Ending default
Code	Medium	C++	Improper initialization of character array
Code	Trivial	C++	Missing break in case Statement(Misra2008)
Code	Low	C++	Missing Case in Switch Statement (Misra2012)
Code	Trivial	C++	Unspecified Size of Empty extern Array
Code	Trivial	C++	Missing ELSE Branch from IF-ELSE
Code	Trivial	C++	Invalid Global Variable Identifier (Misra2012)
Code	Low	C++	Inadequate operator of argument of sizeof
Code	Trivial	C++	Violation of naming rule for pointer variable
Code	Trivial	C++	Too long line
Code	Trivial	C++	Indentation of improper length
Code	Trivial	C++	Multiple Statements in Single Line
Code	Trivial	C++	Violation of rule for white space around keyword
Code	Trivial	C++	Missing white space around binary operator
Code	Trivial	C++	Violation of rule for white space in method call
Code	Trivial	C++	Violation of rule for white space in for statement
Code	Trivial	C++	White space around unary operator
Code	Trivial	C++	Violation of rule for white space in casting
Code	Trivial	C++	Violation of rule for white space around accessing operator
Code	Trivial	C++	Wrong inclusion order

Code	Trivial	C++	Violation of rule for white space around pointer operator
Code	Trivial	C++	Violation of rule for white space around function
Code	Trivial	C++	Violation of rule for white space in parentheses
Code	Trivial	C++	Violation of rule for white space around semicolon
Code	Trivial	C++	Violation of indentation rule for block
Code	Trivial	C++	Missing curly brace in compound statement
Code	Trivial	C++	Missing curly brace in compound statement
Code	Trivial	C++	Violation of rule for white space in while statement
Code	Trivial	C++	Violation of rule for white space in if statement
Code	Trivial	C++	Violation of indentation rule for switch statement
Code	Trivial	C++	Violation of switch Statement Spacing Rule
Code	Trivial	C++	Violation of indentation rule for case and default statements
Code	Trivial	C++	Violation of rule for white space around comma
Code	Trivial	C++	Too many logical operations
Code	Trivial	C++	Violation of naming rule for type
Code	Trivial	C++	Global variable declaration between functions
Code	Trivial	C++	Undeclared local variable at top of block
Code	Trivial	C++	Violation of rule of variable declaration in loop
Code	Trivial	C++	Improper size examination
Code	Trivial	C++	Non-static global variable used within one source file
Code	Trivial	C++	Missing extern in data definition in header file
Code	Trivial	C++	Violation of Type Comment Format
Code	Trivial	C++	Multiple Declarations in Single Line
Code	Low	C++	Unnecessary variable scope
Code	Medium	C++	Violation of format of comments for class
Code	Trivial	C++	Violation of format of function comments
Code	Medium	C++	Violation of format of comments for field
Code	Medium	C++	Violation of format of comments for file
Code	Trivial	C++	Violation of naming rule for variable
Code	Medium	C#	Empty Interface
Code	Medium	C#	Mutable Type in Struct
Code	Low	C#	Use of out Parameter
Code	Trivial	C#	Noncompliant Namespace Name
Code	Trivial	C#	Violation of naming rule for parameter

Code	Trivial	C#	Noncompliant Local Variable Name
Code	Low	C#	Definition of Single Field Class
Code	Low	Java	Immediate Use of Integer Literal
Code	Medium	Java	Throwing of Non-allowed Exception
Code	Trivial	Java	Too Long Line
Code	Trivial	Java	Violation of General Rule of Line-wrapping at Braces
Code	Trivial	Java	Single-variable Expression in Condition
Code	Trivial	Java	Missing ELSE clause
Code	Trivial	Java	Modifiers in Wrong Order
Code	Trivial	Java	Compound Statement without Braces
Code	Trivial	Java	Too Wide Live Range of Variable
Code	Medium	Java	Unused Default in Switch
Code	Trivial	Java	Misuse of Return in if-else Statement
Code	Trivial	Java	Empty while Statement
Code	Trivial	Java	Violation of Documentation Comment Format
Code	Trivial	Java	Violation of rule for white space around If keyword
Code	Trivial	Java	Violation of rule for white space around while keyword
Code	Trivial	Java	Violation of rule for white space around for keyword
Code	Trivial	Java	Violation of rule for white space around do-while keyword
Code	Trivial	Java	Violation of rule for white space around switch keyword
Code	Trivial	Java	Violation of rule for white space around try-catch keyword
Code	Trivial	Java	Misuse of Whitespace for Incremental/Decremental Operator
Code	Trivial	Java	Violation of rule for white space around assert keyword
Code	Medium	Java	Violation of format of comments for class
Code	Medium	Java	Violation of format of comments for field
Code	Medium	Java	Violation of format of general comments
Code	Medium	Java	Violation of format of comments for method
Code	Medium	Java	Violation of format of comments for package
Code	Medium	Java	Violation of format of comments for file
Code	Trivial	Java	Special character used in name

Code	Trivial	Java	Too Long Name
Code	Trivial	Java	Hard coded number
Code	Trivial	Java	Ambiguous order of priority of expressions
Code	Trivial	Java	Too Complicated for Loop Control
Code	Trivial	Java	Missing tab space in declaration
Code	Trivial	Java	Violation of naming rule for parameter
Code	Low	Java	Duplicate Name
Code	Trivial	Java	Violation of format of block comments
Code	Trivial	Java	Violation of format of single line comments
Code	Trivial	Java	Violation of format of comments after source code
Code	Trivial	Java	Missing comment at end of block
Code	Trivial	Java	White space around unary operator
Code	Trivial	Java	Unparenthesized Conditional Operation
Code	Trivial	Java	Improper increment and decrement operators
Code	Trivial	Java	Missing white space around binary operator
Code	Trivial	Java	Multiple Statements in Single Line
Code	Trivial	Java	Violation of indentation rule for block
Code	Trivial	Java	Missing package declaration
Code	Trivial	Java	Missing parentheses in return statement
Code	Trivial	Java	Missing curly brace in if statement
Code	Trivial	Java	Missing finally block
Code	Trivial	Java	Violation of rule for white space in for statement
Code	Trivial	Java	Violation of rule for white space in casting
Code	Trivial	Java	Violation of indentation rule for if statement
Code	Trivial	Java	Multiple Declarations in Single Line
Code	Trivial	Java	Declaration at middle of block
Code	Trivial	Java	Violation of format of curly braces
Code	Trivial	Java	Misformatted Array Declaration
Code	Low	Java	Unsynchronized public method
Code	Trivial	Java	Use of Synchronized Statement
Code	Trivial	Java	Missing curly brace in compound statement
Code	Trivial	Java	Violation of naming rule for constant
Code	Trivial	Java	Violation of naming rule for variable
Code	Trivial	Java	Violation of naming rule for method
Code	Trivial	Java	Violation of naming rule for class
Code	Trivial	Java	Locking or unlocking in loop

Code	Trivial	Java	Reading stream in loop
Code	Trivial	Java	Missing blank line
Code	Trivial	Java	Missing public class or interface
Code	Trivial	Java	Unhandled additional exceptions
Code	Trivial	Java	Violation of rule for white space in method call
Code	Trivial	Java	Missing initialization of local variable
Code	Trivial	Java	Immediate initialization of field
Code	Trivial	Java	Violation of rule for white space in parentheses
Code	Trivial	Java	Violation of rule for white space around keyword
Code	Trivial	Java	Violation of rule of class organization
Code	Trivial	Java	Indentation at Beginning of Source Code
Code	Trivial	Java	Too long line
Code	Trivial	Java	Indentation of improper length
Code	Trivial	Java	Violation of indentation rule for for statement
Code	Trivial	Java	Violation of Method Declaration Rule
Code	Trivial	Java	Violation of rule for white space around accessing operator
Code	Trivial	Java	Violation of Line-wrapping Rule
Code	Trivial	Java	Violation of Line-wrapping Rule between import Statements
Code	Trivial	Java	Missing blank line between method blocks
Code	Trivial	Java	Type-import-on-demand Declaration
Code	Trivial	Java	Violation of indentation rule for class
Code	Trivial	Java	Violation of indentation rule for method
Code	Trivial	Java	Violation of rule of field declaration
Code	Trivial	Java	Violation of indentation rule for field
Code	Trivial	Java	Violation of Semicolon Spacing Rule
Code	Trivial	Java	Violation of Variable Declaration Rule
Code	Trivial	Java	Violation of rule of variable declaration in loop
Code	Trivial	Java	Use of integer type
Code	Trivial	Java	Use of float type
Code	Trivial	Java	Violation of rule for white space in while statement
Code	Trivial	Java	Violation of indentation rule for do statement
Code	Trivial	Java	Violation of rule for white space in if statement
Code	Trivial	Java	Violation of indentation rule for switch statement
Code	Trivial	Java	Violation of switch Statement Spacing Rule

Code	Trivial	Java	Violation of indentation rule for case and default statements
Code	Trivial	Java	Multiple case Branches in Single Line
Code	Trivial	Java	Violation of indentation rule for try statement
Code	Trivial	Java	Violation of catch Statement Spacing Rule
Code	Trivial	Java	Violation of rule for white space around comma
Code	Trivial	Java	Constant of Inner Class
Code	Trivial	Java	Too many logical operations
Code	Trivial	Java	String appended via + and += operator
Code	Trivial	Java	Instance creation in loop
Code	Trivial	Java	Violation of naming rule for package
Code	Trivial	Java	Violation of indentation rule for package
Code	Trivial	Java	Empty branch statement
Code	Trivial	Java	String comparison via equals method
Code	Trivial	Java	Accessing instance
Code	Trivial	Java	Redundant name
Code	Trivial	Java	Violation of indentation rule for comment
Code	Low	JS/TS	Unnecessary Block
Code	Low	JS/TS	Unnecessary Parentheses
Code	Trivial	JS/TS	Missing curly brace
Code	Trivial	JS/TS	Violation of naming rule for variable
Code	Medium	SQL	Missing comment for column in SELECT statement
Code	Trivial	SQL	Violation of naming rule for table
Code	Critical	SQL	Use of Forbidden Table
Code	Critical	SQL	Use of Forbidden Table Column
Code	Medium	ABAP	Use of BREAK-POINT statement
Code	Medium	ABAP	Use of SYSTEM-CALL statement
Code	Medium	ABAP	Use of system C functions
Code	Low	ABAP	CX_ROOT Exception Handling
Code	Medium	ABAP	Missing WHEN OTHERS clause
Code	Low	ABAP	Too Short WHEN Clause
Code	Low	ABAP	Empty catch block
Code	Low	ABAP	Missing ELSE clause
Code	Medium	ABAP	Use of DATA BEGIN OF OCCURS Statement
Code	Medium	ABAP	Use of NOT IN
Code	High	ABAP	Missing WHERE clause in DELETE statement

Code	Medium	ABAP	Use of EXIT and CHECK statements in loop
Code	Medium	ABAP	Use of * in SELECT Statement
Code	Medium	ABAP	Use of native SQL
Code	Low	ABAP	Use of FORM Statement
Code	High	ABAP	Missing WHERE in UPDATE Statement
Code	Low	ABAP	Use of REFRESH FROM TABLE Statement
Code	Low	ABAP	Use of SELECT Statement in Loop
Code	Medium	ABAP	Too deeply nested control flows
Code	Medium	ABAP	Too big file
Code	Medium	ABAP	Nested SELECT Statement
Code	Medium	ABAP	Too many branches in loop
Code	Medium	ABAP	Missing SORT field
Code	Medium	ABAP	Missing ORDER BY clause in SELECT statement
Code	Medium	ABAP	Missing WHERE clause in SELECT statement
Code	Medium	ABAP	Use of Internal Source Code-handling Statement
Code	Medium	ABAP	Use of BYPASSING BUFFER Clause
Code	Medium	ABAP	Use of DISTINCT Operator
Code	Low	ABAP	Duplicated string
Code	Medium	C	Misuse of TRY and CATCH macro
Code	Medium	C	Missing call of commit or rollback functions in transaction
Code	Low	C	Use of Forbidden Restrict Qualifier
Code	Low	C	Use of forbidden tgmath.h header
Code	Medium	C	Inadequate type of argument of scanf
Code	Medium	C	Inadequate size of allocated block
Code	Medium	C	Assignment of parameter to improper type
Code	Medium	C	Assignment of mixed char and integer types
Code	High	C	Missing call of required function
Code	Low	C	Missing function declaration
Code	Medium	C	Misuse of FILE Object
Code	Medium	C	Array-type Parameter Declaration with Static Keyword
Code	Low	C	Modification of Parameter in Call-by-value Function
Code	Trivial	C	Inline Function Declaration without Static
Code	Low	C	Macro Named after Keyword
Code	Trivial	C	One-line Comment Spanning Multiple Lines
Code	Low	C	Useless Label

Code	Low	C	Useless Tag Declaration
Code	Low	C	Useless Macro Declaration
Code	Low	C	Duplicate Declarations of Identifier with External Linkage
Code	Medium	C	Side effect on initializer list
Code	Medium	C	Side effect on designated initializer list
Code	Medium	C	Missing Item in Designator List
Code	Low	C	Use of Compiler-specific Code Extension
Code	Trivial	C	Invalid Return from Preprocessor Conditional Expression
Code	Trivial	C	Duplicate Initializer in Designator List
Code	Low	C	Forbidden exception handling in fenv.h header
Code	Low	C	Use of forbidden function of stdarg.h header
Code	Low	C	Too Long Function Code
Code	Low	C	Block with Too Many Levels of Nesting
Code	Low	C	Out of Range of Constant Argument in Function
Code	Low	C	Too many execution paths
Code	Low	C	Division by Undecidable Value
Code	Trivial	C	Implicit Function Declaration
Code	Medium	C	Write to Read-only File
Code	Medium	C	Use of cnd_wait without Loop
Code	Medium	C	Double-freeing of Thread
Code	Low	C	Disable added language features
Code	Low	C	Validating values passed to <ctype.h> functions
Code	Low	C	Checking Pointer Argument Type Compatibility
Code	Low	C	Handling returned pointer const
Code	Low	C	Do not reuse returned pointers
Code	Low	C	Compare the returned EOF data
Code	Low	C	Check for errors after setting errno
Code	Low	C	Check for invalid errno
Code	Low	C	Useless Assignment
Code	Low	C++	Use of forbidden argument
Code	Medium	C++	Improper value on sting argument
Code	Medium	C++	Use of unspecified argument
Code	Medium	C++	Inadequate variable arguments
Code	Medium	C++	Implicit casting of float

Code	Medium	C++	Unencapsulated assembly language
Code	Low	C++	Identifier with Over 31 Characters
Code	Trivial	C++	Duplicated name in namespace
Code	Medium	C++	Assignment of improper type
Code	High	C++	Replaced character type variable
Code	High	C++	Character value as operand of operation
Code	Medium	C++	Missing explicit casting
Code	Medium	C++	Unused wrapper function
Code	Low	C++	Replaced character type variable
Code	Trivial	C++	Direct use of primitive type
Code	High	C++	Improper type of bit field
Code	Medium	C++	Too small signed integer type
Code	Low	C++	Use of Octal Number
Code	Low	C++	Suspicious octal escape sequence
Code	Low	C++	Function declaration in function
Code	Medium	C++	Unnecessary global variable
Code	Medium	C++	Duplicated external object or function
Code	Medium	C++	Violation of rule of argument usage after function call
Code	Medium	C++	Non-static object or function without external reference
Code	Medium	C++	Missing curly brace in array initialization
Code	Medium	C++	Insufficient initialization value
Code	Medium	C++	Partially initialized enumeration list
Code	Medium	C++	Implicit casting of integer
Code	Medium	C++	Implicit upcasting of integer
Code	Medium	C++	Implicit downcasting of integer
Code	Medium	C++	Mismatched return types of definition and declaration
Code	Medium	C++	Use of forbidden macro
Code	Medium	C++	Implicit casting of float
Code	Medium	C++	Implicit downcasting of float
Code	High	C++	Casting integer to larger type
Code	High	C++	Conversion between signed and unsigned data
Code	Low	C++	Casting float to larger type
Code	Medium	C++	Missing casting for result of bitwise operation
Code	High	C++	Casting from pointer type to integer type
Code	High	C++	Casting from integer type to pointer type

Code	Medium	C++	Casting to pointer type
Code	Medium	C++	Use of forbidden string as argument
Code	Medium	C++	Ambiguous order of priority of expressions
Code	Medium	C++	Side effect on expression
Code	Medium	C++	Side effect on function call
Code	Medium	C++	Side effect on parameter
Code	Medium	C++	Assignment in operands
Code	Low	C++	Use of volatile type variable in complex expression
Code	Medium	C++	Side effect on size examination
Code	Medium	C++	Side effect on function call in wrong order
Code	High	C++	Side effect on logical operation
Code	Medium	C++	Missing specified argument in function
Code	Medium	C++	Opening already opened file
Code	Low	C++	Improper deallocation of array
Code	Low	C++	Use of void parameter type
Code	High	C++	Accessing volatile variable in logical operation
Code	High	C++	Bitwise operation on signed value
Code	High	C++	Shifting on signed value
Code	Medium	C++	Negative operation on unsigned value
Code	Low	C++	Use of comma operator
Code	High	C++	Use of bit representation on floating point value
Code	Medium	C++	Implicit comparison expression
Code	Medium	C++	Use of float type as loop index
Code	Low	C++	Missing update of loop index
Code	Low	C++	Missing condition of loop statement
Code	Low	C++	Modification of variable except loop index in loop
Code	Low	C++	Multiple initializations in loop
Code	Medium	C++	Missing specified argument
Code	Low	C++	Update of loop index in loop
Code	Low	C++	Use of boolean expression in condition of switch statement
Code	Low	C++	Empty switch statement
Code	Low	C++	Variable arguments in function
Code	Trivial	C++	Improper declaration of read-only parameter
Code	Medium	C++	Direct use of function identifier
Code	Medium	C++	Arithmetic operation on general pointer

Code	Medium	C++	Subtraction of pointers of different objects
Code	Medium	C++	Use of literal as argument
Code	Medium	C++	Comparison between pointers of different objects
Code	Medium	C++	Array indexing on pointer type variable
Code	Medium	C++	Assignment between objects from overlapped memory region
Code	High	C++	Assignment to overlapping storage
Code	Low	C++	Use of forbidden standard library functions
Code	Low	C++	Use of forbidden time functions
Code	Trivial	C++	Use of break statement at outside of switch statement
Code	Medium	C++	Non-const function pointer
Code	Medium	C++	Use of variable as argument
Code	Low	C++	Anonymous field in structure and union
Code	Low	C++	Redefinition of pointer type
Code	Low	C++	Lowercase suffix for long
Code	Medium	C++	Improper declaration of flexible array member
Code	High	C++	Use of structure as argument of memcmp
Code	Trivial	C++	Assumed constant value
Code	Medium	C++	Improper association of operators
Code	Medium	C++	Semicolon on same line with statement
Code	Medium	C++	Accessing volatile object through non-volatile reference
Code	Medium	C++	Use of non-variable argument
Code	Low	C++	Modification to Temporary Object
Code	Low	C++	Missing error handling for input function
Code	Medium	C++	Shift and arithmetic operations in single expression
Code	Low	C++	Assumed signed integer representation
Code	Low	C++	Missing casting of floating point return value
Code	Trivial	C++	Non-const string pointer
Code	Medium	C++	Concatenation of different type strings
Code	Medium	C++	Modification of string literal
Code	Low	C++	Missing casting after memory allocation
Code	High	C++	Hard coded data
Code	Medium	C++	Static allocation or copying with variable length member array
Code	Low	C++	Insecure transmission of binary data between systems

Code	Low	C++	Wrong file open mode
Code	Low	C++	Multiple use of ungetc
Code	Low	C++	Inadequate value of argument of fsetpos
Code	Low	C++	Use of errno to check for FILE stream errors
Code	Medium	C++	Missing required qualifier on pointer parameter
Code	Medium	C++	Improper exit method
Code	Medium	C++	Uninitialized errno
Code	Low	C++	Direct use of variable of type time_t
Code	High	C++	Improper assert statement
Code	Low	C++	Too complex switch statement
Code	Medium	C++	Use of forbidden qualifier on pointer parameter
Code	Low	C++	Blocking while holding POSIX lock
Code	Low	C++	Missing curly brace in switch statement
Code	Low	C++	Missing curly brace in if-else statement
Code	Medium	C++	Use of invalid error code
Code	Low	C++	Incomplete structure and union type
Code	High	C++	Use of Union
Code	Low	C++	Non-standard character in include statement
Code	Low	C++	Violation of format of included file name
Code	Low	C++	Defined or undefined macro in block
Code	Low	C++	Use of #undef directive
Code	Medium	C++	Missing call of initialization function on error
Code	Medium	C++	Function-like macro
Code	Medium	C++	Mismatched number of arguments of macro
Code	Low	C++	Preprocess statement in macro
Code	Medium	C++	Missing parentheses around macro argument
Code	Low	C++	Undefined identifier
Code	Low	C++	Multiple # or ## operation
Code	Trivial	C++	Use of # or ## operator
Code	Low	C++	Non-standard format of preprocess statement
Code	Low	C++	Extra character after preprocess statement
Code	Low	C++	Trailing semicolon at preprocess statement
Code	Medium	C++	Improper return value on error
Code	Trivial	C++	Dynamic memory allocation
Code	Trivial	C++	Use of errno
Code	Trivial	C++	Use of offsetof Macro

Code	Trivial	C++	Use of setjmp
Code	Trivial	C++	Use of longjmp
Code	Low	C++	Use of Signal Handling
Code	Trivial	C++	Use of stdio
Code	Low	C++	Use of Wide String
Code	Low	C++	Returning in function of void return type
Code	Low	C++	Arithmetic Operation on Pointer
Code	Low	C++	Use of setlocale
Code	Medium	C++	Signal Handler''s Access to Shared Object
Code	Low	C++	Returning signal handlers
Code	Low	C++	Unused runtime constraint handler
Code	Medium	C++	Use of signal in multi-threaded program
Code	Medium	C++	Overloaded logical operator
Code	Medium	C++	Missing logging on error
Code	Medium	C++	Use of C-style casting
Code	Medium	C++	Use of copy initialization
Code	Medium	C++	Missing special function in complicated class
Code	Medium	C++	Missing special function in dynamically allocated class
Code	Medium	C++	Missing explicit destructor in complex class
Code	Medium	C++	Missing explicit destructor in dynamically allocating class
Code	Trivial	C++	Too complex inline function
Code	Trivial	C++	Non-virtual inline function
Code	Trivial	C++	Too long inline function
Code	Trivial	C++	Use of inline member function
Code	High	C++	Hard coded argument
Code	Medium	C++	Conversion operator for primitive type
Code	Medium	C++	Conversion operator for class type
Code	Trivial	C++	Missing output operator
Code	Low	C++	Non-standard interface
Code	Medium	C++	Missing keyword explicit
Code	Low	C++	Non-public derivation
Code	Medium	C++	Non-virtual destructor for base class
Code	Medium	C++	Dangerous downcasting
Code	Medium	C++	Casting to virtual class
Code	High	C++	Hard coded initialization value

Code	Medium	C++	Missing overriding of overloaded function
Code	Low	C++	Too many parameters
Code	Low	C++	Multiple updates of loop index
Code	Trivial	C++	Mutable condition of loop statement
Code	Low	C++	Multiple entry and exit points
Code	Low	C++	Improper switch statement
Code	Low	C++	Improper variable declaration in for statement
Code	Trivial	C++	Missing suffix for integer constant
Code	Trivial	C++	Missing suffix for floating point constant
Code	Low	C++	Non-const global and static variable
Code	Medium	C++	Use of forbidden number as argument
Code	High	C++	Casting from floating point type to integer type
Code	Low	C++	Implicit casting of argument
Code	Low	C++	Use of global variable
Code	Trivial	C++	Use of using directive
Code	Trivial	C++	Using statement before include statement
Code	Low	C++	Static object declaration in namespace
Code	Trivial	C++	Use of extern
Code	Medium	C++	Use of static specifier in header file
Code	Trivial	C++	Use of auto
Code	Trivial	C++	Use of register keyword
Code	Low	C++	Missing const qualifier
Code	Medium	C++	Use of result of assignment operation
Code	Medium	C++	Mixed signed and unsigned data
Code	Medium	C++	Mixed arithmetic precision
Code	Low	C++	Passing by value
Code	Low	C++	Overloaded numeric and pointer types
Code	Low	C++	Use of default arguments for overloaded function
Code	Low	C++	Unused user defined type on array
Code	Low	C++	Uninitialized pointer after release
Code	Low	C++	Comment in macro
Code	Low	C++	Non-blank line at end of source code
Code	Low	C++	Uppercase in included file name
Code	Low	C++	Improper constant definition
Code	Low	C++	Use of array type
Code	Low	C++	Use of Digraph

Code	Low	C++	Use of non-standard header file
Code	Low	C++	Inefficient copying object in container
Code	Low	C++	Use of copy constructor
Code	Low	C++	Dynamic array allocation
Code	Medium	C++	Unnecessary reallocation
Code	Low	C++	Improper passing of vector to C-style function
Code	High	C++	Modification of key
Code	Low	C++	Mixed iterator types
Code	Medium	C++	Non-pure predicate function
Code	Medium	C++	Use of forbidden type
Code	Medium	C++	Use of STL algorithms
Code	Low	C++	Use of auto_ptr
Code	Low	C++	Definition of inline member function in class declaration
Code	Low	C++	Missing exception handling for memory allocation
Code	Low	C++	Too high level of dereferencing
Code	Low	C++	Pointer dereference operation in macro
Code	Low	C++	Duplicated function or variable name between files
Code	Low	C++	Missing declaration of extern variable
Code	Low	C++	Unused user defined type on pointer
Code	Medium	C++	Use of universal character in macro concatenation
Code	Low	C++	Missing relationship between constants
Code	Low	C++	Use of compound literal address in loop
Code	Low	C++	Non-unique mutually visible identifier
Code	Low	C++	Use of non-volatile variable in signal handler
Code	Medium	C++	Use of reserved identifier
Code	Medium	C++	Side effect on macro argument
Code	Medium	C++	Unused user defined type
Code	High	C++	Accessing variable via incompatible pointer
Code	High	C++	Improper casting of integer
Code	Medium	C++	Copying FILE Object
Code	Medium	C++	Redefined errno
Code	Low	C++	Parameter name in function prototype
Code	Medium	C++	Virtual function call in constructor and destructor
Code	Low	C++	Just one of overloaded new and delete
			Returning non-const value from overloaded postfix

Code	Low	C++	operator
Code	Low	C++	Recursive call during initialization of static object
Code	Low	C++	Unused user defined type on member variable
Code	Low	C++	Deleting or casting pointer of incomplete class
Code	Medium	C++	Float operation on non-float value
Code	Low	C++	Predicate function with non-static field
Code	Medium	C++	Specified bound of character array
Code	Medium	C++	Missing check of sameness on copy assignment operation
Code	Low	C++	Use of unaligned pointer
Code	Medium	C++	Throwing exception in destructor
Code	Medium	C++	Accessing field in handler of constructor
Code	Critical	C++	Wrong exception handling order
Code	Low	C++	Throwing exception during deallocation
Code	Low	C++	Use of forbidden array type
Code	Medium	C++	Non-virtual destructor
Code	Medium	C++	Throwing exception on copy assignment operation
Code	Low	C++	Missing initialization in constructor
Code	Medium	C++	Wrong initialization order in constructor
Code	Low	C++	Public constructor for abstract class
Code	Medium	C++	Overloaded non-virtual function
Code	Low	C++	Mismatched default values
Code	Low	C++	Non-protected copy constructor
Code	Medium	C++	Use of not-allowed type
Code	Low	C++	Implicit virtual function
Code	Low	C++	Return of non-const handle from const member function
Code	Trivial	C++	Non-abstract base class
Code	Low	C++	Too many base classes
Code	Low	C++	Member binary operator
Code	Low	C++	Improper overloading of subscript operator
Code	Low	C++	Too many execution paths
Code	Low	C++	Throwing non-class type object
Code	Medium	C++	Catching exception by reference
Code	Low	C++	Use of forbidden compound type
Code	Low	C++	Use of increment operator on boolean value

Code	Low	C++	Casting from integer type to enumeration type
Code	Low	C++	Implicit conversion of class template
Code	Medium	C++	Conflicting method in class template
Code	Medium	C++	Derived class object in container for base class object
Code	Low	C++	Comparing size of container with zero
Code	Low	C++	Use of STL container as public base class
Code	Low	C++	Use of forbidden pointer type
Code	High	C++	Creation of container of auto_ptr
Code	Medium	C++	Use of vector of boolean type
Code	Low	C++	Violation of naming rule for identifier
Code	Trivial	C++	Extern variable and function declaration in source file
Code	Low	C++	Use of magic number
Code	Low	C++	Use of forbidden user defined type
Code	Low	C++	Missing void keyword
Code	Low	C++	Public class member variable
Code	Low	C++	Undeclared constructor
Code	Low	C++	Undeclared destructor
Code	Low	C++	Undeclared copy constructor
Code	Low	C++	Undeclared assignment operator
Code	Medium	C++	Improperly overloaded assignment operator
Code	Low	C++	Non-member asymmetric operator
Code	High	C++	Missing check of array index
Code	Low	C++	Non-friend symmetric operator
Code	Medium	C++	Improper memory reallocation
Code	Medium	C++	Modification of constant value
Code	Medium	C++	Mismatched type in ternary operation
Code	High	C++	Insufficient allocated memory
Code	Low	C++	Use of #pragma directive
Code	Low	C++	Empty statement without comment
Code	Low	C++	Name in standard library
Code	Low	C++	Use of friend function and class
Code	Medium	C++	Improper comparison of condition
Code	Medium	C++	Missing Casting to larger type
Code	High	C++	Casting from void pointer type to non-void pointer type
Code	Low	C++	Missing reset of string on failure

Code	Medium	C++	Missing check of array length
Code	Medium	C++	Use of void return type
Code	Low	C++	Ineffective statement
Code	Medium	C++	Pointer dereference operation in type definition
Code	Low	C++	Lost of precision in integer casting
Code	Medium	C++	Missing check of debug mode
Code	Low	C++	Use of macro as instance
Code	Medium	C++	Missing class member assignment
Code	Medium	C++	Missing class member assignment
Code	Medium	C++	Misuse of rename
Code	Low	C++	Performing operation on device
Code	Low	C++	Wrong type of error
Code	Low	C++	Duplicated header file name
Code	High	C++	Improper conversion of string token
Code	Medium	C++	Missing exception handling for floating point errors
Code	Medium	C++	Use of pointer operation on field
Code	Low	C++	Improper file opening and creation
Code	Low	C++	Storing returned pointer to non-const variable
Code	Medium	C++	Variable declaration in loop
Code	Low	C++	Code deletion during compiler optimization
Code	Low	C++	Use of va_arg on indeterminated value
Code	Low	C++	Violation of constraints of extern inline function
Code	Medium	C++	Non-standard character
Code	Medium	C++	Unused Parameter
Code	Trivial	C++	Use of do-while statement
Code	Low	C++	Backward goto Statement
Code	Low	C++	Jump between Blocks
Code	Low	C++	Use of Variable-length Array in Struct
Code	Medium	C++	Use of == operator on floating point values
Code	Trivial	C++	Illegible comment
Code	Trivial	C++	Different comment lengths
Code	Trivial	C++	Violation of format of comments
Code	Trivial	C++	Comment around code
Code	Low	C++	Unused user defined type on parameter
Code	Low	C++	Returning non-constant value
Code	Medium	C++	Unused user defined type for return type

Code	Medium	C++	Use of character type
Code	Medium	C++	Including duplicated header file
Code	Medium	C++	Unused static variable
Code	Low	C++	Use of bit field type
Code	Medium	C++	Use of float type
Code	Medium	C++	Use of extern or static specifier
Code	Medium	C++	Use of forbidden qualifier
Code	Trivial	C++	Too many functions
Code	Medium	C++	Improper file input and output
Code	Low	C++	Missing definition of declared function
Code	Medium	C++	Use of bitwise operation on boolean type
Code	Low	C++	Multiple declarations of different types in single line
Code	Low	C++	Including forbidden header file
Code	Low	C++	Exposure of structure of type
Code	Medium	C++	Returning null array
Code	Medium	C++	Assignment of local address to upper scope
Code	Low	C++	Inappropriate including method of file
Code	Medium	C++	Executable statement before first case statement
Code	Low	C++	Use of absolute path for inclusion
Code	Low	C++	Missing header file
Code	Trivial	C++	Improper structure alignment
Code	Trivial	C++	Padded structure
Code	Low	C++	Too big structure
Code	Medium	C++	Unreachable case statement
Code	Medium	C++	Mismatched number of variable arguments
Code	Medium	C++	Virtual function call without qualifier
Code	Medium	C++	Use of dynamic type in constructor and destructor
Code	Medium	C++	Unhandled exception in destructor
Code	Low	C++	Unused exception handling
Code	Low	C++	Use of remainder operation
Code	Low	C++	Returning parameter address
Code	Medium	C++	Static casting on virtual base class pointer
Code	Medium	C++	Unhandled exception in main
Code	Medium	C++	Dividing on signed value
Code	Medium	C++	Missing comparison of return value
Code	High	C++	Mismatched operand types

Code	Low	C++	Conflicting storage class
Code	Low	C++	Uninitialized pointer as read-only parameter
Code	Medium	C++	Inconsistent global variable declaration
Code	High	C++	Mismatched character type
Code	High	C++	Mismatched reallocated type
Code	Medium	C++	Missing break statement in case statement
Code	Low	C++	Use of multidimensional array
Code	Low	C++	Declaration at middle of block
Code	Low	C++	Use of function in condition of if statement
Code	Low	C++	String comparison via == and != operators
Code	Medium	C++	Assignment of character type to integer type
Code	Medium	C++	Non-decimal integer constant
Code	High	C++	Invalidated Iterator
Code	Medium	C++	Accessing local variable address
Code	Medium	C++	Mismatched number of arguments
Code	High	C++	Reused tokenized string
Code	Medium	C++	Symbolic link race condition
Code	Low	C++	Direct assignment of physical address to function pointer
Code	High	C++	Asynchronous thread termination
Code	Critical	C++	Assigning instead of comparing
Code	Critical	C++	Assigning instead of comparing
Code	Trivial	C++	Non-virtual pure function
Code	Trivial	C++	Improper initialization of pure virtual function
Code	Medium	C++	Mismatched types of function definition and declaration
Code	Medium	C++	Inadequate value of argument of memcpy
Code	Trivial	C++	Redundant name
Code	Medium	C++	Missing return type
Code	Low	C++	Empty while Statement
Code	Low	C++	Use of Hexadecimal Escape Sequence
Code	Medium	C++	Trailing semicolon at macro definition
Code	High	C++	Use of improper macro
Code	High	C++	Replacing secure function
Code	High	C++	Unbracketed macro including multiple statements
Code	Medium	C++	Duplicated macro

Code	Medium	C++	Incorrect Byte Order
Code	Critical	C++	Locking already locked resource
Code	Critical	C++	Double unlocked resource
Code	High	C++	Trylocking already locked resource
Code	Medium	C++	Inadequate value of file descriptor
Code	High	C++	Too large stack size
Code	Trivial	C++	Ineffective function call
Code	Low	C++	Class with Template Does Not Have Copy Assignment Operator Definition
Code	Low	C++	Function Declaration without External Side-effect
Code	Trivial	C++	Non-constant Operands of Bitwise Operator
Code	Low	C++	Diamond Problem with Non-virtual Inheritance
Code	High	C++	Assignment of Local Variable Address to Wide-scoped Variable
Code	Low	C++	Exception Handler Missed in Main Function
Code	Trivial	C++	Use of cstdio Header
Code	Trivial	C++	Use of cstring Header
Code	Trivial	C++	Use of ctime Header
Code	Medium	C++	Throwing of Null Exception
Code	Low	C++	Throwing of Pointer Type Exception
Code	Trivial	C++	Empty throw Statement
Code	Low	C++	Use of Null Pointer for Integer
Code	Low	C++	enum Declaration of Bit Field
Code	Medium	C++	Return of Argument Passed by Reference
Code	Low	C++	Overloading of address operator
Code	Trivial	C++	Use of Virtual Inheritance
Code	Low	C++	Both Virtual and Non-Virtual Inheritance in Same Hierarchy
Code	Trivial	C++	Function Never Used
Code	Trivial	C++	Throwing of Non-allowed Exception
Code	Low	C++	Bit Field Declaration of Non-allowed Data Type
Code	Trivial	C++	Use of Assembly Command without asm
Code	Low	C++	Unused Parameter in Virtual Function
Code	Trivial	C++	Comment Present in Section of Code
Code	Low	C++	Exception Thrown from Constructor or Destructor
Code	Low	C++	Change of Static Field in Copy Constructor

Code	Low	C++	Class with Template Does Not Have Copy Constructor Definition
Code	Trivial	C++	Uninstantiated Template
Code	Low	C++	Ambiguous Call to Explicitly Specialized Function
Code	Trivial	C++	Ambiguous Call for Template Specialization Function
Code	Low	C++	Throw Involves Another Throw
Code	Trivial	C++	Different Exception Specification
Code	Trivial	C++	Non-member Template Function Definition in Namespace
Code	Low	C++	Ambiguous Call to Template Base Method
Code	Trivial	C++	String Allocation to Invalid Type
Code	Medium	C++	Changed loop control variable in condition or increment or decrement clause
Code	Medium	C++	Non-boolean Loop Control Variable
Code	Low	C++	Use of continue in Unusually-formed for Loop
Code	Low	C++	Internal-Linkage Function without Static Storage Class
Code	Medium	C++	Missing Catch for Explicitly Thrown Exception
Code	Low	C++	Non-const Handle Returned to Class Data
Code	Trivial	C++	Token Mismatch in Redeclaration
Code	Trivial	C++	Misuse of unsigned int-type Suffix
Code	Medium	C++	Missing Termination of Case
Code	Medium	C++	Use of C Style Memory-related Function
Code	Medium	C++	Infinite loop
Code	Medium	C++	Misuse of socket
Code	Trivial	C++	Use of forbidden function
Code	Medium	C++	Comparing function pointer instead of return value
Code	Medium	C++	Adjustment of Pointer Size for Pointer Operation
Code	Trivial	C++	Function declaration in function
Code	Medium	C++	Ambiguous Call to Template Function
Code	Medium	C++	Using Casting to Remove const Qualifier
Code	Medium	C++	Violation of va_start Macro Limitation
Code	High	C++	Unfreed Memory after Use of tss_create()
Code	Medium	C++	Reference to Atomic Variable Twice in Expression
Code	Low	C++	Failure of atomic_compare_exchange_weak()
Code	Medium	C++	Conflicting Storage Classes
Code	Medium	C++	Unused return of standard libraries

Code	Low	C++	Incorrect Vararg Type
Code	Medium	C++	Comparison on Floating-point Objects
Code	Medium	C++	Use of Incorrect Integer Precision
Code	Medium	C++	Manipulation of Another Thread''s Mutex
Code	Medium	C++	Local Variable Shared between Threads
Code	Low	C++	Violation of naming rule for enumeration type
Code	Low	C++	Missing Include Guard in Class Definition Header
Code	Trivial	C++	Violation of Operator Spacing Rule
Code	Low	C++	Use of constexpr in switch
Code	Medium	C++	Range Error in Math Function
Code	Medium	C++	Destruction of Locked Mutex
Code	Medium	C++	Shared Object with Expired Storage Duration
Code	Medium	C++	Occurrence of Deadlock
Code	Medium	C++	Misuse of Condition Variable
Code	High	C++	Already Owned Pointer
Code	Medium	C++	Use of volatile Qualifier for Reference Type
Code	Low	C++	Return from noreturn Function
Code	Medium	C++	Array with Polymorphism
Code	High	C++	Increase/Decrease Iterator by more than One
Code	High	C++	Call to mutex_unlock() Function without Exception Handling
Code	Medium	C++	Use of Maximum Buffer Size
Code	Low	C++	Use of Decrement Operator on Boolean
Code	Trivial	C++	Missing initialization of local variable
Code	Medium	C++	Missing #define in header file
Code	Medium	C++	Unreachable code due to string comparison
Code	High	C++	Remainder operation on negative value
Code	High	C++	Assignment of negative value to unsigned type
Code	Medium	C++	Division by zero
Code	Medium	C++	Missing check on division by zero
Code	High	C++	Returning local variable address
Code	Medium	C++	Releasing memory in stack
Code	High	C++	Missing return statement
Code	Low	C++	Overlapped memory region
Code	High	C++	Shifting which exceeds bit width
Code	High	C++	Shifting on negative value

Code	Medium	C++	Inadequate value of length argument
Code	Low	C++	Redundant condition
Code	Low	C++	Unreachable code
Code	Low	C++	Unused value
Code	High	C++	Dangerous casting of function pointer
Code	Medium	C++	Violation of format of general comments
Code	Trivial	C++	Use of label statement
Code	Medium	C++	Use of forbidden function in loop
Code	Medium	C#	Missing Braces in while Statement
Code	Medium	C#	Missing Braces in for Statement
Code	Low	C#	Instantiation of string
Code	Medium	C#	Missing default case in switch statement
Code	Medium	C#	Missing break statement in case statement
Code	Medium	C#	Jump Statement at End of Loop
Code	Medium	C#	Use of Equals on Variable
Code	Medium	C#	Empty string used in + operator
Code	High	C#	Hard coded IP
Code	Medium	C#	Assignment to multiple variables
Code	Trivial	C#	Instance creation in loop
Code	Medium	C#	Too deeply nested if statements
Code	Medium	C#	Throwing NullPointerException
Code	Medium	C#	Use of == operator on floating point values
Code	Medium	C#	String appended via += operator
Code	Medium	C#	Comparison of Collection Size against 0
Code	Medium	C#	Unused local variable
Code	Medium	C#	Unused parameter
Code	Medium	C#	Unused private field
Code	Medium	C#	Throwing new exception
Code	Medium	C#	Unpulsed Object
Code	Low	C#	Assigning instead of comparing
Code	Medium	C#	Empty while Statement
Code	Medium	C#	Use of Invalid Access Modifier on sealed Class
Code	Medium	C#	Use of toString on Array
Code	Trivial	Java	Missing Call to Superclass Method in Android Activity
Code	Trivial	Java	Missing Necessary Call in Override of onMeasure
Code	Low	Java	Immediate Assignment of Literal to Specific Variable

Code	High	Java	Dangerous downcasting of object in collection
Code	Medium	Java	Suspicious Method Name
Code	High	Java	Misuse of equals and ==
Code	High	Java	Incorrect Overriding
Code	High	Java	Modification of static field in non-static method
Code	Medium	Java	Use of notify
Code	Low	Java	Use of wait in synchronized block without control
Code	Low	Java	Use of notify in synchronized block
Code	Trivial	Java	Violation of Specified Exception Handling Rule
Code	Low	Java	Use of sleep or yield at outside of control block
Code	Medium	Java	Use of == operator on floating point values
Code	Low	Java	Inappropriate Removal of Object in Collection
Code	Low	Java	Access to Private Member of Nesting Class
Code	Medium	Java	Unreleased Lock
Code	Medium	Java	Unnotified object
Code	Low	Java	Unnecessary casting on iterator variable
Code	Low	Java	Implicit casting in function call
Code	Low	Java	Implicit casting in assignment
Code	Low	Java	Implicit casting in return
Code	Low	Java	Nested try Statement
Code	Low	Java	Returning private field of outer class
Code	Low	Java	Return of String Using Constructor
Code	Trivial	Java	Use of Runtime.exec
Code	Trivial	Java	Use of Finalizer on Exit
Code	Medium	Java	Use of raw type
Code	Trivial	Java	Use of deprecated API
Code	Trivial	Java	Use of synchronized method in loop
Code	Trivial	Java	Access to Array with Fixed Index in Loop
Code	Trivial	Java	Break statement using label
Code	Trivial	Java	Use of volatile
Code	Trivial	Java	Use of short Type
Code	Trivial	Java	Use of Public Field
Code	Low	Java	Unused import statement
Code	Trivial	Java	Use of public field in method
Code	Medium	Java	Assignment to multiple variables
Code	Critical	Java	Assigning instead of comparing

Code	Low	JS/TS	Invalid Operand Type
Code	Low	JS/TS	Use of with Statement
Code	Trivial	JS/TS	Inappropriate Equality Comparison Operator
Code	Low	JS/TS	Assignment in operands
Code	Low	JS/TS	Bitwise Operation
Code	Medium	JS/TS	Modified native object
Code	Low	JS/TS	Improper variable declaration in for statement
Code	Trivial	JS/TS	Unreachable code
Code	High	Obj-C	Symbolic link race condition
Code	High	Obj-C	Misuse of Apple APIs
Code	High	Obj-C	Passing nil to @synchronized directive
Code	Critical	Obj-C	Releasing non-instance object
Code	Medium	Obj-C	Misuse of CFNumberCreate
Code	High	Obj-C	Use of float type as loop index
Code	Medium	Obj-C	Use of uninitialized value in binary operation
Code	Medium	Obj-C	Use of uninitialized value in array
Code	Medium	Obj-C	Assignment of Uninitialized Variable
Code	Medium	Obj-C	Use of uninitialized value in conditional statement
Code	Medium	Obj-C	Return of Uninitialized Variable
Code	Medium	Obj-C	Misuse of CFErrorRef
Code	High	Obj-C	Misuse of malloc
Code	High	Obj-C	Invalid malloc Argument
Code	High	Obj-C	Assignment of Stack Variable Address
Code	Medium	Obj-C	Invalid Argument for String Function
Code	Medium	Obj-C	Misuse of CFRetain, CFRelease and CFMakeCollectable
Code	Low	Obj-C	Uninitialized self
Code	Low	Obj-C	Misuse of Secure Keychain APIs
Code	Low	Obj-C	Unused variable
Code	Low	Obj-C	Logical error in function call and messaging
Code	Medium	Obj-C	Inadequate value of length argument
Code	Low	Obj-C	Use of IPv4-specific API
Code	Low	Obj-C	Hard coded IP
Code	Medium	Obj-C	Unlocked mutex
Code	High	Obj-C	Buffer overflow
Code	Low	Obj-C	Lost of precision in assignment
			Missing check of return value from

Code	Low	Obj-C	pthread_mutex_lock
Code	Low	Obj-C	Use of == operator on floating point values
Code	Trivial	Obj-C	Use of forbidden function
Code	Medium	Python	Missing Exception Handling for ConnectionError
Code	Medium	Python	Missing Exception Handling for BlockingIOError
Code	Medium	Python	Missing Exception Handling for FileExistError
Code	Critical	SQL	Missing WHERE clause in SELECT statement
Code	Critical	SQL	Nonexistent Table
Code	Critical	SQL	Nonexistent Column
Code	Low	SQL	Unused Table
Code	Low	SQL	Unused Column
Code	High	SQL	Invalid Number Type
Code	High	SQL	Longer Input than Column Size
Code	Critical	SQL	Invalid GROUP BY Clause
Code	High	SQL	Null into Non-nullable Column
Code	Critical	SQL	Invalid Date Format
Code	Critical	SQL	Syntax Error
Code	Critical	SQL	Invalid Table Name
Code	High	SQL	Null into PRIMARY KEY Column
Code	Critical	SQL	Unselected Column Appears in ORDER BY
Code	Critical	SQL	Mismatched number of columns and values in INSERT query
Code	Critical	SQL	Missing WHERE clause in DELETE and UPDATE statement
Code	Trivial	Swift	Not Updated Left-hand Operand of Assignment
Code	Trivial	Swift	Ternary Operator Returning Fixed Value
Code	Trivial	Swift	Redundant Condition Check
Code	Trivial	Swift	System Function Call
Code	Trivial	Swift	Binary Operation with Identical Operands
Code	Trivial	Swift	Extra Code after Branching Statement
Code	Trivial	Swift	Branching Statement Not Included in Conditional Statement
Code	Trivial	Swift	Relational Operation on Floating-point Value
Code	Trivial	Swift	Forced-unwrapped Optional
Code	Trivial	Swift	Operator Precedence Change
Code	Trivial	Swift	Implicitly Unwrapped Optional

Code	Low	Swift	No Explaining Message for Fingerprint Request
Code	Trivial	Etc.	Use of Specified Keyword
Code	Trivial	VBS	Use of forbidden function
Code	Medium	VBS	Missing check of empty value
Code	Low	VBS	Use of with Statement
Code	Medium	VBS	Missing return statement
Code	Medium	Java	return Statement in catch Block
Code	High	JS/TS	Non-binary Transaction
Code	Low	Swift	Implicit Access to Class
Code	High	VB.Net	Unrestricted Action
Code	Medium	VB.Net	Disabling of Header Check
Code	High	VB.Net	Disabling of View State MAC
Code	Low	VB.Net	Execution with Impersonated Credentials
Code	Medium	VB.Net	Leftover debug code
Code	Low	VB.Net	Storage of Non-Serializable Object as HttpSessionState Attribute
Code	High	VB.Net	Parent Model without Required Attribute
Code	Medium	VB.Net	Setting of Persistent Permission
Code	High	VB.Net	Untrusted Model
Code	Medium	ABAP	Use of SELECT statement with dynamic clauses
Code	Low	C	Use of Variable-length Array
Code	Low	C	Macro Named after Reserved Word
Code	Medium	C	Forbidden recursive call
Code	Low	C	Useless Type Declaration
Code	Low	C	Passing of Signed to Character Function
Code	Medium	C	getchar Function Return of Invalid Type
Code	Medium	C	Null Dereference Resulting from Allocation Failure
Code	Medium	C	Dereference before null check
Code	Low	C++	Use of Trigraph
Code	Medium	C++	Too many initialization values
Code	High	C++	Lost of const qualification in casting
Code	High	C++	Lost of volatile qualification in casting
Code	Low	C++	Missing error detection
Code	Medium	C++	Misuse of assert statement
Code	Low	C++	Use of PTHREAD_MUTEX_NORMAL type mutex lock
Code	High	C++	Use of vfork

Code	Low	C++	Use of pthread_kill
Code	Medium	C++	Use of signal functions
Code	High	C++	Use of sizeof on array type parameter
Code	High	C++	Improper conversion between numeric types
Code	High	C++	Inadequate size of allocated memory
Code	Medium	C++	Printing file to temporary or public directory
Code	High	C++	Use of system
Code	High	C++	Exit on atexit handler
Code	High	C++	Use of unsafe function in signal handler
Code	Medium	C++	Use of deprecated API
Code	Medium	C++	Premature thread termination
Code	High	C++	Use of sizeof on pointer type
Code	Medium	C++	Throwing overly broad exceptions
Code	High	C++	Wrong order in privilege relinquishment
Code	High	C++	Insecure privilege relinquishment
Code	High	C++	Integer Underflow
Code	High	C++	Invalid downcasting of integer
Code	High	C++	Use of released resource
Code	High	C++	Mismatched buffer size
Code	Medium	C++	Too Small Path Buffer
Code	Low	C++	Critical public Variable without const
Code	Medium	C++	Use of macros in wrong order
Code	High	C++	Misuse of asctime
Code	High	C++	Integer Overflow
Code	High	C++	Use of Dangerous Function
Code	Medium	C++	Use of unsafe multi-byte string function
Code	Medium	C++	Infinite recursive call
Code	Medium	C++	Mismatch in Number between Format Specifiers and Arguments
Code	Trivial	C++	Empty branch statement
Code	Medium	C++	Access to Bit-field by Multiple Threads
Code	Medium	C++	Alteration of Standard Namespace
Code	Low	C++	Exception Thrown in Handler Registered with atexit()
Code	High	C++	Container Overflow
Code	High	C++	Dynamically allocated buffer overflow
Code	Low	C++	Modifying Source Object within Copy Operation

Code	Medium	C++	Use of C Standard Library to Target Unsuitable Class Object
Code	Low	C++	Public static member variable undeclared to const
Code	Medium	C++	Memory Leak
Code	High	C++	Resource leak
Code	High	C++	Double freed memory
Code	High	C++	Use of freed memory
Code	High	C++	Uninitialized value
Code	Medium	C++	Null dereference
Code	Medium	C++	Missing null check
Code	High	C++	Returning freed memory
Code	Medium	C++	Attempting delete on dynamically allocated memory
Code	Medium	C++	Attempting delete[] on dynamically allocated memory
Code	Medium	C++	Attempting delete on memory allocated by new[]
Code	Medium	C++	Attempting free on memory allocated by new[]
Code	Medium	C++	Attempting delete[] on memory allocated by new
Code	Medium	C++	Attempting free on memory allocated by new
Code	High	C++	Double freed resource
Code	Low	C++	Missing call of specified library function
Code	Medium	C++	Missing call of library function with specified parameter
Code	Medium	C++	Missing call of required library function with required argument
Code	Medium	C++	Missing call of required function
Code	Medium	C++	Misuse of malloc
Code	Medium	C++	Misuse of process APIs
Code	Medium	C++	Misuse of resource APIs
Code	Medium	C++	Misuse of signal APIs
Code	Medium	C++	Misuse of TP APIs
Code	Medium	C++	Use of temporary file-related function
Code	Medium	C++	Misuse of timer APIs
Code	Medium	C++	Use of printf
Code	Trivial	C++	Accessing array via subscript operator
Code	Medium	C++	Use of umask before calling fopen
Code	Medium	C++	Accessing member of structure multiple times
Code	Medium	C#	Division by zero

Code	Medium	C#	Null dereference
Code	Medium	C#	Dereference of Null Return Value
Code	Medium	C#	Missing null check
Code	Medium	C#	Use of Application.Exit
Code	Medium	C#	Passing of String to IndexOf
Code	Medium	C#	Use of Monitor.Pulse
Code	High	C#	Resource leak
Code	Medium	C#	Null return value dereference in standard library
Code	Low	C#	Empty finally Block
Code	Trivial	C#	Use of forbidden interface
Code	Medium	C#	Missing Serializable Attribute
Code	Medium	C#	Null Check via Equals Method
Code	Medium	C#	Just one of defined GetHashCode and Equals
Code	Low	C#	Misuse of SqlClientPermission
Code	Medium	C#	Hard coded file separation character
Code	Medium	Java	Allowing Javascript on Android
Code	Medium	Java	Granting URI permission on Android
Code	Medium	Java	Broadcasting intents on Android
Code	Medium	Java	Using JavaScript in Android
Code	High	Java	Class loading hijacking on Android
Code	Medium	Java	Class loading hijacking due to absolute paths on Android
Code	Medium	Java	Bypassing permission checking on Android
Code	Low	Java	Use of AWT or Swing
Code	Low	Java	Use of Java IO
Code	Medium	Java	Android Security Alert Notification
Code	Medium	Java	Cross-site scripting by escapeXml
Code	Medium	Java	Missing Broadcaster Permission in Android
Code	Medium	Java	Cross-site scripting in Android
Code	High	Java	Use of public static final Array
Code	High	Java	Overloading of equals
Code	High	Java	Use of equals between Arrays
Code	Medium	Java	Missing check of return value from string methods
Code	High	Java	Direct Management of Connections
Code	High	Java	Direct Use of Sockets
Code	Medium	Java	Race Condition for Database Connection

Code	Medium	Java	Resource leak
Code	Medium	Java	Division by zero
Code	Medium	Java	Null dereference
Code	Medium	Java	Null return value dereference
Code	Medium	Java	Missing null check
Code	High	Java	Missing call of super class method
Code	High	Java	Use of Mutable Field in compareTo
Code	High	Java	Use of Mutable Field in hashCode
Code	Medium	Java	Null return value dereference in standard library
Code	High	Java	Improper exception handling
Code	High	Java	Invalid Modifier for serialPersistentFields
Code	Medium	Java	Missing synchronization in overriding
Code	High	Java	Double-checked locking
Code	High	Java	Use of server socket in EJB
Code	Medium	Java	Infinite recursive call
Code	Medium	Java	Empty catch block
Code	High	Java	Incorrect overriding of hashCode and equals
Code	High	Java	Direct Use of Threads
Code	High	Java	Use of System.exit
Code	Trivial	Java	Use of forbidden function
Code	Low	Java	Division without strictfp qualifier
Code	Medium	Java	Missing break statement in case statement
Code	Critical	Java	Globally accessible file
Code	Trivial	JS/TS	Use of forbidden function
Code	Medium	JS/TS	Synchronized transaction
Code	Medium	JS/TS	Empty catch block
Code	Medium	Kotlin	Empty catch block
Code	Medium	Kotlin	Null return value dereference
Code	Trivial	Kotlin	Use of forbidden function
Code	High	Obj-C	Use of system
Code	High	Obj-C	Insecure privilege relinquishment
Code	Critical	Obj-C	Null dereference
Code	Critical	Obj-C	Double String Formatting
Code	Medium	Obj-C	Buffer overflow by CFArrayGetValueAtIndex
Code	Critical	Obj-C	Division by zero
Code	High	Obj-C	Invalid Size of Variable-length Array

Code	Medium	Obj-C	Misuse of mktemp
Code	Medium	Obj-C	Misuse of vfork
Code	Medium	Obj-C	Misuse of CFArrayCreate
Code	Medium	Obj-C	Misuse of getpw
Code	Medium	Obj-C	Misuse of gets
Code	Medium	Obj-C	Misuse of random
Code	Medium	Obj-C	Misuse of strcpy
Code	Medium	Obj-C	Misuse of Unix APIs
Code	Medium	Obj-C	Infinite recursive call
Code	High	Obj-C	Insecure privilege control
Code	High	Obj-C	Insecure privilege reset
Code	High	Obj-C	Use of Anonymous LDAP Binding
Code	Critical	Obj-C	Use of freed memory
Code	High	Obj-C	Use of Dangerous Function
Code	Critical	Obj-C	Memory Leak
Code	Medium	PHP	Infinite recursive call
Code	Medium	PHP	Empty catch block
Code	Medium	PHP	Catching overly broad exceptions
Code	Medium	PHP	Null dereference
Code	High	PHP	Resource leak
Code	High	PHP	Use of released resource
Code	Trivial	PHP	Use of forbidden function
Code	Medium	Python	Hardcoded Path Separator
Code	Trivial	Python	Use of forbidden function
Code	Critical	Rust	Improper Cargo Setting
Code	Critical	Rust	Use of unsafe
Code	High	Rust	Wrapping Potential Overflow
Code	Trivial	Rust	Use of try!
Code	Critical	Rust	Use of panic Inducing Function
Code	Medium	Rust	Access to Misformatted Array
Code	Critical	Rust	Division by zero
Code	High	Rust	Use of forget
Code	High	Rust	Memory Leak
Code	Medium	Rust	Improper Release of ManuallyDrop
Code	Critical	Rust	Inaccurate Return of Pointer
Code	Low	Rust	Use of Uninitialized Memory

Code	Medium	Rust	Use of Incompatible Type
Code	Medium	Swift	Transmission via HTTP Protocol
Code	Trivial	Swift	Use of forbidden function
Code	Medium	Swift	Infinite recursive call
Code	High	Etc.	Enabled debuggable option in Android manifest
Code	Medium	Etc.	Enabled sharedUserId option in Android manifest
Code	Medium	Etc.	Enabled exported option in Android manifest
Code	High	Etc.	Account information in source code
Code	High	Etc.	Credit card information in source code
Code	High	Etc.	Email information in source code
Code	High	Etc.	Foreigner registry number in source code
Code	High	Etc.	IP information in source code
Code	High	Etc.	Resident registration number in source code
Code	High	Etc.	Passport number in source code
Code	High	Etc.	Phone number in source code
Code	High	Etc.	Driver license information in source code
Code	Trivial	VB.Net	Use of forbidden function
Code	Trivial	VB.Net	Empty branch statement
Code	Medium	ASP	Cross-site scripting
Code	High	ASP	SQL Injection
Code	High	ASP	Command injection
Code	High	ASP	LDAP Injection
Code	High	ASP	Path Traversal
Code	Critical	ASP	XQuery Injection
Code	Medium	ASP	HTTP Response Splitting
Code	High	ASP	XPath Injection
Code	High	ASP	Resource injection
Code	Medium	ASP	Redirection to untrusted site
Code	Medium	ASP	Cleartext transmission of sensitive information
Code	Medium	ASP	Cross site scripting via error messages
Code	High	ASP	Reliance on DNS lookups in security decisions
Code	High	ASP	Weak password requirements
Code	Medium	ASP	Persistent cookie
Code	Medium	ASP	Exposure of system information
Code	Medium	ASP	Untrusted Regex
Code	High	ASP	Unrestricted File Upload

Code	Medium	ASP	Storing Unencrypted Password
Code	Medium	C	Missing check of return value
Code	High	C++	Insecure file identification
Code	Critical	C++	Incorrect permission assignment in file creation
Code	High	C++	Duplicated environment variable name
Code	Low	C++	Reliance on Environment Pointer
Code	Low	C++	Use of in-band error indicator
Code	Medium	C++	Missing check of symbolic links
Code	Low	C++	Use of non-static variable as argument of putenv
Code	Low	C++	File descriptor race condition
Code	Low	C++	Misuse of signal handler
Code	Low	C++	Use of signal in interruptible signal handler
Code	Low	C++	Library race condition
Code	Low	C++	Information leak due to structure padding
Code	High	C++	Missing limitation for string length
Code	Low	C++	Writing sensitive information to disk
Code	Medium	C++	Missing check of input length
Code	Medium	C++	Violation of privilege to use functions in Windows
Code	High	C++	Weak hash
Code	Medium	C++	Improper encryption
Code	High	C++	Improper random number generation
Code	Medium	C++	Leftover debug code
Code	Medium	C++	Multiple signal assignments to same handler
Code	Medium	C++	Use of RSA Algorithm without OAEP
Code	Medium	C++	Heap inspection
Code	Medium	C++	Catching overly broad exceptions
Code	Medium	C++	Switch statement race condition
Code	Medium	C++	Improper pointer scaling
Code	Medium	C++	Insufficient exponent for cryptographic key
Code	High	C++	Buffer overflow dynamically assigned in wrong condition
Code	High	C++	Dynamically Assigned Buffer Underflow
Code	High	C++	Dynamically Assigned Buffer Underflow Caused by Improper Condition
Code	Medium	C++	TOCTOU race condition on filename
Code	Low	C++	Private collection returned by public method

Code	Low	C++	Storage of External Data in Private Collection
Code	Medium	C++	Invalid umask Argument
Code	Low	C++	Storage of External Data in Private Field
Code	Low	C++	Critical Public Variable
Code	Medium	C++	Reallocation of aligned memory
Code	High	C++	Command injection of call on system function
Code	High	C++	Missing null termination character
Code	High	C++	Misuse of unbounded input
Code	High	C++	Buffer overflow by function
Code	High	C++	Buffer overflow by function in wrong condition
Code	High	C++	Field buffer overflow by function
Code	High	C++	Field buffer overflow by function in wrong condition
Code	High	C++	Buffer overflow dynamically assigned by function
Code	High	C++	Buffer overflow dynamically assigned by function in wrong condition
Code	High	C++	Field Buffer Underflow from Function
Code	High	C++	Field Buffer Underflow from Function Caused by Improper Condition
Code	Medium	C++	Missing changing working directory
Code	High	C++	Buffer Underflow from Function
Code	High	C++	Buffer Underflow from Function Caused by Improper Condition
Code	High	C++	Dynamically Assigned Buffer Underflow from Function
Code	High	C++	Dynamically Assigned Buffer Underflow from Function Caused by Improper Condition
Code	Medium	C++	Reliance on DNS lookups in security decisions
Code	Medium	C++	Use of getlogin in multi-threaded program
Code	High	C++	Hard coded password
Code	High	C++	Hard coded user name
Code	Medium	C++	Weak cryptographic algorithm
Code	Medium	C++	Insecure privilege control
Code	Medium	C++	Insecure privilege reset
Code	High	C++	Insufficient cryptographic key size
Code	Medium	C++	Inappropriate RSA Padding
Code	High	C++	Hard coded salt
Code	High	C++	Weak cryptographic algorithm of password

Code	Medium	C++	Multiple bindings to same port
Code	Critical	C++	Incorrect permission assignment for critical resource
Code	High	C++	SQL Injection
Code	High	C++	Path Traversal
Code	High	C++	Command injection
Code	High	C++	LDAP Injection
Code	High	C++	Resource injection
Code	Critical	C++	External Control of System or Configuration Setting
Code	High	C++	Insecure direct object reference
Code	High	C++	Password in comment
Code	High	C++	Improper authorization
Code	Medium	C++	Cleartext storage of password
Code	High	C++	Hardcoded cryptographic key
Code	Medium	C++	Use of Tainted Value
Code	Medium	C++	Empty catch block
Code	High	C++	Unrestricted File Upload
Code	Medium	C++	Redirection to untrusted site
Code	Critical	C++	XQuery Injection
Code	High	C++	XPath Injection
Code	High	C++	Mismatched resource release method
Code	Medium	C++	Persistent cookie
Code	High	C++	Improper Reference of XML Entity
Code	Medium	C++	Weak Server Certificate
Code	High	C++	Code Injection of Return Address
Code	High	C++	Missing login control
Code	Low	C++	Missing authentication
Code	High	C++	Missing password recovery control
Code	High	C++	Transmission of Key Security Information and Vehicle Information in Plain Text
Code	High	C++	Use of Low-security Encryption Algorithm
Code	High	C++	Missing Message ID when Generating MAC
Code	Low	C++	Grammatically Ambiguous Declaration
Code	Low	C++	Missing Exception Safety
Code	Medium	C++	Missing Exception Handling
Code	Critical	C++	Reuse of Moved-from Object
Code	Medium	C++	Use of placement new Operator on Improper Object

Code	Medium	C++	Unhandled Exception from Statically Scoped Object Declaration
Code	Critical	C++	Pointer-to-member Operator for Non-existent Member Access
Code	Medium	C++	Casting of Out-of-range Enum Value
Code	High	C++	Buffer overflow
Code	High	C++	Buffer overflow in wrong condition
Code	High	C++	Field buffer overflow in wrong condition
Code	High	C++	Field buffer overflow
Code	High	C++	Buffer overflow in range-bound copy
Code	Medium	C++	Cross-site scripting
Code	High	C++	Weak password requirements
Code	High	C++	Use of Hash without Salt
Code	High	C++	Missing limit of login attempts
Code	Medium	C++	TOCTOU race condition
Code	Medium	C++	Missing check of return value
Code	High	C++	Download of code without integrity check
Code	High	C++	Reliance on untrusted inputs in security decisions
Code	High	C++	Field Buffer Underflow Caused by Improper Condition
Code	High	C++	Field Buffer Underflow
Code	High	C++	Buffer Underflow
Code	Trivial	C++	Improper sequential memory allocation
Code	Trivial	C++	Using multithreading without synchronization in the Singleton pattern
Code	High	C++	Buffer Underflow Caused by Improper Condition
Code	High	C++	Format string injection
Code	High	C++	Exposure of system information
Code	High	C#	Hardcoded user name and password
Code	Medium	C#	Non-ASCII character used in file name and path
Code	High	C#	Integer Overflow
Code	High	C#	Missing XML validation
Code	High	C#	Use of Hash without Salt
Code	High	C#	Public data assigned to private array
Code	High	C#	Private collection returned by public method
Code	Medium	C#	Name-based Type Check
Code	High	C#	Improper authorization

Code	High	C#	Missing login control
Code	High	C#	Missing authentication for critical function
Code	High	C#	Password in comment
Code	High	C#	XSLT Injection
Code	High	C#	Data Leak between Sessions
Code	High	C#	Hardcoded cryptographic key
Code	High	C#	Download of code without integrity check
Code	Medium	C#	TOCTOU race condition
Code	High	C#	Incorrect permission assignment for critical resource
Code	Medium	C#	Weak Security Checks on Serialization Implementation
Code	High	C#	Blank Password
Code	High	C#	Hardcoded HMAC Private Key
Code	Medium	C#	Hardcoded Initialization Vector
Code	High	C#	Empty HMAC Private Key
Code	High	C#	Key Derivation Function with Blank Password
Code	High	C#	Key Derivation Function with Hardcoded Password
Code	Critical	C#	Use of Unsafe DLL
Code	Medium	C#	Missing header file
Code	High	C#	Disabling of EnableViewStateMac
Code	Medium	C#	Use of Predictable Salt
Code	High	C#	Key Derivation Function with Insecure Iteration Count
Code	Medium	C#	Inappropriate RSA Padding
Code	High	C#	Use of Anonymous LDAP Binding
Code	High	C#	Non-HttpOnly cookie
Code	High	C#	Use of cookie with overly broad domain
Code	High	C#	Use of Cookie with Overly Broad Path
Code	High	C#	Use of Cryptographic Algorithm in ECB Mode
Code	High	C#	Short Signature Key
Code	High	C#	Sensitive Information Exposal in UI Loading
Code	High	C#	Hardcoded Private Key in Symmetric Key Algorithm
Code	High	C#	Hardcoded Password
Code	High	C#	Comparison with Hardcoded Password
Code	High	C#	Exception Information Exposure
Code	High	C#	Dynamic code manipulation
Code	High	C#	Weak XML Transformer
Code	High	C#	Server-side Request Forgery

Code	High	C#	Inappropriate Signature
Code	Medium	C#	Weak Server Certificate
Code	High	C#	Deserialization of Untrusted Data
Code	High	C#	Cross-site Request Forgery
Code	Medium	C#	Catching NullPointerException
Code	Medium	C#	Empty catch block
Code	Critical	C#	Use of HtmlInputHidden
Code	Medium	C#	Exposure of system information
Code	Medium	C#	Catching overly broad exceptions
Code	Low	C#	Insecure logging
Code	High	C#	Weak hash
Code	High	C#	Weak cryptographic algorithm
Code	High	C#	Insufficient cryptographic key size
Code	High	C#	Improper random number generation
Code	High	Dart	Hardcoded API keys
Code	High	Dart	Hardcoded Credential
Code	High	Dart	Hard-coded email addresses
Code	High	Dart	Hardcoded IP addresses
Code	High	Dart	Weak Hash
Code	High	Dart	Vulnerable Random Number
Code	High	Dart	Weak cryptographic algorithm
Code	High	Dart	Insufficient RSA Encryption Key Length
Code	High	Dart	API keys in comments
Code	High	Dart	Credentials in comments
Code	High	Dart	Email addresses in comments
Code	High	Dart	IP addresses in comments
Code	High	Go	Cross-site Script (HTML)
Code	High	Go	Cross-site Script (JS)
Code	High	Go	Cross-site Script (template)
Code	High	Go	Cross-site Script (etc.)
Code	High	Go	Cross-site Script (URL)
Code	High	Go	SQL Injection
Code	High	Go	Weak hash
Code	High	Go	Vulnerable Random Number
Code	Medium	Go	Error Message Information Exposed
Code	Medium	Go	Improper Cookie Flag Setting

Code	Low	Go	Vulnerable Package
Code	High	Go	Hard coded password
Code	High	Go	Improper Network Settings
Code	High	Go	Improper SSH Key Settings
Code	High	Go	Server-side Request Forgery
Code	High	Go	Integer Conversion Overflow
Code	Medium	Go	Slowloris Attack
Code	High	Go	Improper File Privilege Setting
Code	High	Go	Path Traversal
Code	Medium	Go	Improper TLS Settings
Code	High	Go	Insufficient RSA Encryption Key Length
Code	High	HTML	Exposure of password
Code	High	HTML	Inclusion of Script from External Site
Code	Medium	HTML	Password Field with Autocomplete Enabled
Code	Medium	Java	Unlimited appending to collection
Code	Medium	Java	Deletion of item of collection in loop
Code	Critical	Java	Non-static serializable inner class
Code	High	Java	Improper implementation of Externalizable interface
Code	Critical	Java	Untrusted input in AccessController.doPrivileged
Code	Medium	Java	Cycle in class initialization
Code	Medium	Java	Cycle in class initialization
Code	Medium	Java	Non-final loop index
Code	Medium	Java	Expression in assert statement
Code	Medium	Java	Lost of precision in casting
Code	Medium	Java	Use of raw type collection
Code	High	Java	Unsafe mutable class
Code	High	Java	Inheritance of sensitive class
Code	High	Java	Log injection
Code	Medium	Java	Insecure File Extraction through ZipInputStream
Code	Medium	Java	Non-ASCII character used in file name and path
Code	Medium	Java	Untrusted Regex
Code	Critical	Java	Non-character used in input validation
Code	Medium	Java	Missing NaN Check for Real Number
Code	High	Java	Comparing with string representation of floating point values
Code	Medium	Java	Dangerous downcasting

Code	Medium	Java	Direct use of mutable inputs and internal components
Code	Medium	Java	Blocked external process on IO buffer
Code	Medium	Java	Inadequate integer value of argument of write
Code	Medium	Java	Use of Untrusted File Link
Code	Medium	Java	Incorrect Serialization Order
Code	Medium	Java	Memory and resource leakages during serialization
Code	Critical	Java	Use of default automatic signature verification
Code	Medium	Java	Improper use of return value from readInt
Code	Critical	Java	Throwing exception in constructor
Code	Medium	Java	Improper comparison between key objects
Code	Medium	Java	Improper URL comparison
Code	Medium	Java	Improper Restoration on Failure
Code	Medium	Java	Use of same thread pool
Code	Medium	Java	Not reinitialized ThreadLocal field
Code	High	Java	Partially initialized object
Code	Critical	Java	Improper use of return value from reading
Code	Critical	Java	Incomplete Static Initializer Block
Code	High	Java	Use of system environment variables
Code	Critical	Java	Use of ReflectPermission
Code	High	Java	Comparing class name
Code	High	Java	Use of parameter in assert statement
Code	High	Java	Critical public method without final modifier
Code	High	Java	Increased accessibility of method
Code	Critical	Java	Overridable method call in clone method
Code	Medium	Java	Overriding public static method
Code	Medium	Java	Unhandled exception in finally block
Code	High	Java	Non-volatile shared field between threads
Code	Medium	Java	Non-thread-safe method chaining
Code	Medium	Java	Use of non-atomic data type
Code	High	Java	Use of reusable object as lock instance
Code	High	Java	Use of class object in synchronization
Code	High	Java	Use of high-level concurrent object as lock instance
Code	Medium	Java	Initialization from another collection
Code	Medium	Java	Non-synchronized static field
Code	Critical	Java	Deadlock
Code	Medium	Java	Waiting while holding lock

Code	Medium	Java	Infinitely waiting thread
Code	Medium	Java	Abrupt thread termination
Code	Medium	Java	Unused thread pool
Code	Medium	Java	Uninterruptible thread in synchronized block
Code	Medium	Java	Improper creation of thread pool
Code	Medium	Java	Thread execution in constructor
Code	Medium	Java	Publishing before initialization
Code	Medium	Java	Creation of thread in static initialization block
Code	High	Java	Misuse of delete for File instance
Code	High	Java	Misuse of deleteOnExit
Code	Critical	Java	Exposure of buffer
Code	Medium	Java	Missing Size Argument for read() Method
Code	Medium	Java	Use of big endian only methods
Code	High	Java	Exposure of field of serialized class
Code	High	Java	Improper implementation of Serializable interface
Code	Medium	Java	Overridable method call in readObject method
Code	Critical	Java	Missing privilege check
Code	Critical	Java	Use of reflection
Code	Critical	Java	Missing call of super class method in getPermissions method
Code	Critical	Java	Importing untrusted class
Code	Medium	Java	Missing infinity check of float input
Code	Medium	Java	Cross site scripting via attributes in request
Code	Critical	Java	Violation of Trust Boundary
Code	High	Java	Generating predictable random value
Code	High	Java	Comparison of Boxed Primitives
Code	Medium	Java	Missing Visibility of Shared immutable Object
Code	High	Java	Use of Socket Class
Code	High	Java	Dynamic code manipulation
Code	High	Java	SpEL Expression Unchecked
Code	High	Java	OGNL Expression Unchecked
Code	High	Java	Weak XML Transformer
Code	Medium	Java	Weak Server Certificate
Code	High	Java	Server-side Request Forgery
Code	High	Java	Deserialization of Unsafe Jackson
Code	High	Java	Deserialization of Unsafe XStream

Code	Medium	Java	Insecure Hostname Unchecked in Android
Code	Medium	Java	Unnormalized String before Validation
Code	Medium	Java	XML Insertion
Code	Medium	Java	Simultaneous Bitwise and Arithmetic Operations
Code	Low	Java	Use of Shift Operator
Code	High	Java	Unvalidated Method Argument
Code	Medium	Java	Incomplete validate Method Definition
Code	Low	Java	Inheriting Validation Class
Code	Medium	Java	Use of sync Primitives
Code	Low	Java	Multiple Buffer Wrappers on Single Byte or Character Stream
Code	Low	Java	Reuse of Public Identifier in Java Standard Library
Code	High	Java	Information Exposure via Java Runtime Error Message
Code	Medium	Java	Sensitive information in log
Code	Medium	Java	Servlet Action after Committed Response
Code	Medium	Java	Logging sensitive information in Android
Code	Critical	Java	Missing Canonicalization before File Exchange through Content Provider
Code	Medium	Java	Missing Normalization before Validation
Code	Medium	Java	Failure to Check Permission on Geolocation API
Code	Medium	Java	XML External Entity Attack
Code	Medium	Java	Use of mutable Field within equals
Code	Medium	Java	Non-serializable Object Stored in Session
Code	High	Java	Unwrapped Native Method
Code	Critical	Java	Security Check for Untrusted Sources
Code	High	Java	Deserialization of Untrusted Data
Code	Trivial	Java	Sensitive Information Included in Comments
Code	High	Java	Direct Use of Unsafe JNI
Code	High	Java	Hardcoded user name and password
Code	High	Java	Insufficient session expiration
Code	High	Java	Insecure direct object reference
Code	High	Java	Dynamic Class Loading
Code	High	Java	Weak cryptographic algorithm
Code	High	Java	SQL Injection
Code	Low	Java	Private collection returned by public method
Code	Medium	Java	Setting up missing J2EE error pages

Code	Trivial	Java	Validating disabled Struts
Code	Trivial	Java	Validation and form field mismatches
Code	High	Java	Validating Unsafe Reflection Input
Code	Medium	Java	Asynchronous access to shared data
Code	Trivial	Java	Using Direct Class Loader in EJB Environment
Code	Trivial	Java	finalize() declared as public
Code	Trivial	Java	Storing objects that cannot be serialized
Code	Trivial	Java	Using public fields in ActionSupport
Code	Trivial	Java	Incorrect autoboxing and unboxing within loops
Code	Trivial	Java	Invalid integer bit shift operations
Code	Medium	Java	Using non-thread-safe singletons
Code	High	Java	Weak password requirements
Code	High	Java	Data Leak between Sessions
Code	Medium	Java	Leftover debug code
Code	High	Java	Nested Class Containing Sensitive Data
Code	High	Java	Critical public variable without final modifier
Code	High	Java	SQL Injection via JDO API
Code	High	Java	SQL Injection via J2EE Persistence API
Code	High	Java	SQL Injection via Hibernate
Code	High	Java	Command injection
Code	High	Java	LDAP Injection
Code	High	Java	Resource injection
Code	High	Java	Path Traversal
Code	Medium	Java	HTTP Response Splitting
Code	Critical	Java	External Control of System or Configuration Setting
Code	Medium	Java	Redirection to untrusted site
Code	Critical	Java	XQuery Injection
Code	High	Java	XPath Injection
Code	Medium	Java	Persistent cookie
Code	Medium	Java	Cross-site scripting
Code	Medium	Java	DOM based cross site scripting
Code	Medium	Java	Direct dynamic code evaluation
Code	High	Java	Reliance on DNS lookups in security decisions
Code	High	Java	Cross site request forgery
Code	High	Java	Exposure of password by memory dump
Code	High	Java	Insufficient cryptographic key size

Code	High	Java	Hardcoded cryptographic key
Code	Medium	Java	Inappropriate RSA Padding
Code	High	Java	Hard coded salt
Code	High	Java	Improper random number generation
Code	High	Java	Exposure of password in address bar
Code	High	Java	Multiple bindings to same port
Code	High	Java	Insecure Cookie
Code	High	Java	Incorrect permission assignment for critical resource
Code	High	Java	Integer Overflow
Code	High	Java	Missing authentication for critical function
Code	High	Java	Improper authorization
Code	Medium	Java	Storing Unencrypted Password
Code	High	Java	Password in comment
Code	High	Java	Use of Hash without Salt
Code	High	Java	Download of code without integrity check
Code	Medium	Java	TOCTOU race condition
Code	High	Java	Public data assigned to private array
Code	High	Java	Reliance on untrusted inputs in security decisions
Code	Medium	Java	Exposure of system information
Code	High	Java	Format string injection
Code	Medium	Java	Cross site scripting via error messages
Code	Medium	Java	Cleartext transmission of sensitive information
Code	High	Java	Password in servlet comment
Code	High	Java	Unrestricted File Upload
Code	High	Java	Missing password recovery control
Code	High	Java	Missing login control
Code	Low	Java	Exposure of administration page
Code	High	Java	Information Leak through Privileged Block
Code	Low	Java	Exposure of Dangerous Method
Code	Low	Java	Missing input validation
Code	Low	Java	Replacing email address
Code	Low	Java	Missing authentication
Code	Low	Java	Insecure password recovery
Code	Medium	Java	Information Leak through Android
Code	Critical	Java	Untrusted Data in Privileged Block
Code	Medium	Java	SSI Injection

Code	Medium	Java	Infinite loop
Code	Medium	JS/TS	Cross-site scripting (ExpressJS)
Code	High	JS/TS	Command injection
Code	Medium	JS/TS	DOM based cross site scripting
Code	Low	JS/TS	Improper property value
Code	Medium	JS/TS	Remote code execution
Code	Medium	JS/TS	Information leak from local storage to session storage
Code	Medium	JS/TS	Information leak from session storage to local storage
Code	Medium	JS/TS	Cross document messaging
Code	High	JS/TS	SQL Injection
Code	High	JS/TS	Use of external data for file creation
Code	High	JS/TS	Predictable database name
Code	High	JS/TS	Hard coded password
Code	High	JS/TS	Empty password
Code	High	JS/TS	Broken access control on databases
Code	High	JS/TS	Command injection
Code	Medium	JS/TS	Direct dynamic code evaluation
Code	Medium	JS/TS	Assignment to innerHTML
Code	Medium	JS/TS	Redirection to untrusted site
Code	Critical	JS/TS	XHR Injection
Code	Medium	JS/TS	Use of localStorage
Code	Medium	JS/TS	Log injection
Code	Medium	JS/TS	Cross-site scripting
Code	High	JS/TS	Insecure MiUpdater
Code	Medium	JS/TS	Leftover debug code
Code	High	JS/TS	Transaction by GET
Code	Critical	JS/TS	Transaction Injection
Code	Medium	JS/TS	Transaction using cleartext
Code	High	JS/TS	Dataset of sensitive information
Code	Medium	JS/TS	Cleartext storage of sensitive information
Code	High	JS/TS	Registry Leak
Code	Medium	JS/TS	Redirection to untrusted site through dialog
Code	High	JS/TS	TOBESOFT Platform OS Command Injection
Code	Low	JS/TS	Too long string for quicktabstextfont attribute
Code	Medium	JS/TS	Broken access control on Azure
Code	Medium	JS/TS	HTTP Response Splitting

Code	Medium	JS/TS	Sensitive Information Exposal
Code	High	JS/TS	JSON injection
Code	Medium	JS/TS	Use of Forbidden Logger
Code	Medium	JS/TS	TOBESOFT Platform Log Manipulation
Code	High	JS/TS	TOBESOFT Platform Path Manipulation
Code	Medium	JS/TS	Exposure of system information
Code	High	JS/TS	Dynamic code manipulation
Code	High	JS/TS	Server-side Request Forgery
Code	High	JS/TS	Inappropriate Signature
Code	Medium	JS/TS	Weak Server Certificate
Code	High	JS/TS	Deserialization of Untrusted Data
Code	High	JS/TS	Cross-site Request Forgery
Code	Medium	JS/TS	Use of uncaughtException
Code	High	JS/TS	Regular expression Denial of Service
Code	High	JS/TS	Strict Mode
Code	High	JS/TS	SQL Injection (mysql)
Code	High	JS/TS	SQL Injection (PostgreSQL)
Code	High	JS/TS	SQL Injection (noSQL)
Code	Low	JS/TS	Improper Cookie Flag Setting
Code	Medium	JS/TS	HTTP Security Header Settings
Code	Medium	JS/TS	Improper HTTP Header
Code	High	JS/TS	Use of weak cryptographic algorithm
Code	Medium	JS/TS	Cross Site Script (dangerouslySetInnerHTML)
Code	High	JS/TS	Use of Vulnerable Function (findDomNode)
Code	Medium	JS/TS	URL Manipulation
Code	Medium	JS/TS	Script Injection
Code	High	JS/TS	SQL Injection (ORM)
Code	Medium	JS/TS	Cross-site Script (VanillaJS)
Code	High	JS/TS	Improper XML External Entity Reference
Code	Medium	JS/TS	One-way Hash Function Used Without Salt
Code	High	JS/TS	Using Encryption Keys of Insufficient Size
Code	High	JS/TS	Path Traversal
Code	High	JS/TS	Use of Insufficient Random Value
Code	Medium	JS/TS	Infinite Loop or Recursive Function
Code	Medium	JS/TS	Exposure of error message information
Code	Medium	JS/TS	Debug Code Not Removed

Code	High	JS/TS	Resource injection
Code	High	JS/TS	Code Injection
Code	High	Kotlin	SQL Injection
Code	High	Kotlin	Weak cryptographic algorithm
Code	High	Kotlin	Path Traversal
Code	High	Kotlin	Command injection
Code	High	Kotlin	Unrestricted File Upload
Code	Medium	Kotlin	Redirection to untrusted site
Code	High	Kotlin	XPath Injection
Code	High	Kotlin	LDAP Injection
Code	High	Kotlin	Format string injection
Code	High	Kotlin	Missing Authentication for Critical Function
Code	High	Kotlin	Improper authorization
Code	High	Kotlin	Incorrect permission assignment for critical resource
Code	Medium	Kotlin	Sensitive Information Storage in Plaintext
Code	Medium	Kotlin	Cleartext transmission of sensitive information
Code	High	Kotlin	Hardcoded Password
Code	High	Kotlin	Insufficient cryptographic key size
Code	High	Kotlin	Improper random number generation
Code	High	Kotlin	Hardcoded Encryption Key
Code	High	Kotlin	Weak password requirements
Code	Medium	Kotlin	Persistent cookie
Code	High	Kotlin	Password in comment
Code	High	Kotlin	Use of Hash without Salt
Code	High	Kotlin	Missing login control
Code	Medium	Kotlin	TOCTOU race condition
Code	Medium	Kotlin	Infinite loop
Code	Medium	Kotlin	Catching overly broad exceptions
Code	High	Kotlin	Data Leak between Sessions
Code	Medium	Kotlin	Leftover debug code
Code	Medium	Kotlin	Exposure of system information
Code	High	Kotlin	Private collection returned by public method
Code	High	Kotlin	Public data assigned to private array
Code	High	Kotlin	Reliance on DNS lookups in security decisions
Code	High	Kotlin	Dynamic code manipulation
Code	High	Kotlin	Weak XML Transformer

Code	High	Kotlin	Server-side Request Forgery
Code	High	Kotlin	Inappropriate Signature
Code	Medium	Kotlin	Weak Server Certificate
Code	High	Kotlin	Deserialization of Untrusted Data
Code	Medium	Kotlin	Insecure Hostname Unchecked in Android
Code	High	Lua	Hardcoded Credential
Code	High	Lua	Hard-coded email addresses
Code	High	Lua	Hardcoded IP addresses
Code	High	Lua	Hardcoded API keys
Code	High	Lua	Weak Hash
Code	High	Lua	Weak random numbers
Code	High	Lua	Insufficient RSA encryption key length
Code	High	Lua	Credentials in comments
Code	High	Lua	Email addresses in comments
Code	High	Lua	IP addresses in comments
Code	High	Lua	API keys in comments
Code	Medium	Lua	Debug Code Not Removed
Code	High	Lua	Weak cryptographic algorithm
Code	Medium	Obj-C	Redirection to untrusted site
Code	Low	Obj-C	Misuse of signal handler
Code	Medium	Obj-C	Missing check of return value
Code	Medium	Obj-C	TOCTOU race condition
Code	Low	Obj-C	Password in comment
Code	High	Obj-C	Buffer overflow in string copy
Code	Medium	Obj-C	Cross-site scripting
Code	Medium	Obj-C	Generation of Dangerous Temp File
Code	High	Obj-C	Command injection
Code	Medium	Obj-C	Empty catch block
Code	Medium	Obj-C	Catching overly broad exceptions
Code	Medium	Obj-C	Reliance on DNS lookups in security decisions
Code	High	Obj-C	Sensitive Information Storage in Plaintext
Code	Critical	Obj-C	XQuery Injection
Code	High	Obj-C	XPath Injection
Code	High	Obj-C	LDAP Injection
Code	High	Obj-C	Reliance on untrusted cookies in security decisions
Code	Medium	Obj-C	Cleartext transmission of sensitive information

Code	High	Obj-C	Download of code without integrity check
Code	High	Obj-C	Use of Cookie with Overly Broad Path
Code	High	Obj-C	Use of cookie with overly broad domain
Code	High	Obj-C	Storage of Sensitive Data in Persistent Cookie
Code	High	Obj-C	Empty Salt
Code	Medium	Obj-C	Use of Null Salt
Code	High	Obj-C	Improper random number generation
Code	Low	Obj-C	Exposure of system information
Code	Medium	Obj-C	Leftover debug code
Code	High	Obj-C	Missing login control
Code	High	Obj-C	Weak password requirements
Code	High	Obj-C	Missing authentication
Code	Medium	Obj-C	HTTP Response Splitting
Code	High	Obj-C	Integer Overflow
Code	High	Obj-C	Improper Reference of XML Entity
Code	Medium	Obj-C	Weak Server Certificate
Code	High	Obj-C	Deserialization of Untrusted Data
Code	Medium	Obj-C	Printing file to temporary or public directory
Code	Medium	Obj-C	Weak SSL Certificate
Code	Medium	Obj-C	Transfer by GET
Code	Medium	Obj-C	Transmission via HTTP Protocol
Code	High	Obj-C	Empty password
Code	High	Obj-C	Hard coded password
Code	Medium	Obj-C	Use of SMS API
Code	High	Obj-C	Hardcoded cryptographic key
Code	High	Obj-C	Weak hash
Code	High	Obj-C	Insufficient cryptographic key size
Code	High	Obj-C	Weak cryptographic algorithm
Code	High	Obj-C	Format string injection
Code	Critical	Obj-C	Log injection
Code	High	Obj-C	Path Traversal
Code	High	Obj-C	Resource injection
Code	High	Obj-C	SQL Injection
Code	High	Obj-C	Insecure Reflection
Code	Medium	PHP	Header manipulation
Code	High	PHP	Command injection

Code	High	PHP	Remote code execution
Code	Medium	PHP	Cookie set to base path
Code	High	PHP	Weak hash
Code	High	PHP	Unrestricted File Upload
Code	Medium	PHP	Redirection to untrusted site
Code	Critical	PHP	XQuery Injection
Code	High	PHP	XPath Injection
Code	High	PHP	LDAP Injection
Code	High	PHP	Cross site request forgery
Code	High	PHP	Reliance on untrusted inputs in security decisions
Code	High	PHP	Format string injection
Code	High	PHP	Missing authentication for critical function
Code	High	PHP	Improper authorization
Code	High	PHP	Weak cryptographic algorithm
Code	Medium	PHP	Cleartext storage of sensitive information
Code	Medium	PHP	Cleartext transmission of sensitive information
Code	High	PHP	Hard coded password
Code	High	PHP	Insufficient cryptographic key size
Code	High	PHP	Improper random number generation
Code	High	PHP	Hardcoded cryptographic key
Code	High	PHP	Weak password requirements
Code	High	PHP	Missing login control
Code	Medium	PHP	Error information leak
Code	Medium	PHP	Leftover debug code
Code	High	PHP	Reliance on DNS lookups in security decisions
Code	High	PHP	Password in comment
Code	Medium	PHP	HTTP Response Splitting
Code	Medium	PHP	Cookie of sensitive information
Code	High	PHP	Use of Hash without Salt
Code	High	PHP	Weak XML Transformer
Code	High	PHP	Server-side Request Forgery
Code	Medium	PHP	Weak Server Certificate
Code	High	PHP	Deserialization of Untrusted Data
Code	High	PHP	Overly Permissive CORS Policy
Code	High	PHP	Transfer by GET
Code	High	PHP	Enabled allowed_url_fopen option

Code	High	PHP	Enabled allowed_url_include option
Code	Medium	PHP	Disabled session.cookie_secure option
Code	High	PHP	Disabled cgi.force_redirect option
Code	Medium	PHP	Disabled safe_mode option
Code	High	PHP	Enabled file_uploads option
Code	High	PHP	Enabled magic_quotes_gpc option
Code	High	PHP	Enabled magic_quotes_runtime option
Code	High	PHP	Enabled magic_quotes_sybase option
Code	Medium	PHP	Enabled register_globals option
Code	Medium	PHP	Enabled display_errors option
Code	Medium	PHP	Exposure of system information
Code	High	PHP	Missing open_basedir Setting
Code	Medium	PHP	Missing safe_mode_exec_dir Setting
Code	Medium	PHP	Cookie setting with base domain
Code	Medium	PHP	Cookie setting with base path
Code	Medium	PHP	Persistent cookie
Code	Medium	PHP	Disabled session.cookie_httponly option
Code	High	PHP	Enabled session.use_trans_sid option
Code	High	PHP	Excessive session timeout on CakePHP
Code	Medium	PHP	Information leak on CakePHP
Code	Medium	PHP	Cleartext transmission of cookies
Code	Medium	PHP	Transmission via HTTP Protocol
Code	Medium	PHP	Cross-site scripting
Code	High	PHP	Resource permission manipulation
Code	High	PHP	SQL Injection
Code	High	PHP	Path Traversal
Code	Critical	PHP	External variable modification
Code	Medium	PHP	Log injection
Code	High	Prop	Password Hardcoded in Configuration File
Code	High	Prop	Blank Password
Code	High	Python	SQL Injection
Code	High	Python	Code Injection
Code	High	Python	Path Traversal and Resource Injection
Code	Medium	Python	Cross-site Script (HTML)
Code	High	Python	Command injection
Code	High	Python	Malicious File Type Upload

Code	Medium	Python	URL Automatic Redirection to Untrusted Site
Code	High	Python	Improper XML External Entity Reference
Code	High	Python	XML Injection
Code	High	Python	LDAP Injection
Code	High	Python	Cross-site Request Forgery
Code	High	Python	Server-side Request Forgery
Code	Medium	Python	Generation of Dangerous Temp File
Code	High	Python	Invalid umask Argument
Code	Medium	Python	Direct dynamic code evaluation
Code	Medium	Python	Log Manipulation
Code	High	Python	SMTP Command Injection
Code	Medium	Python	Memcached Injection
Code	High	Python	Path Traversal
Code	Critical	Python	External Control of System or Configuration Setting
Code	High	Python	Manipulation of django File Response
Code	Medium	Python	HTTP Response Splitting
Code	High	Python	Reliance on Untrusted Inputs in a Security Decision
Code	High	Python	Format string injection
Code	Low	Python	Information Exposure by django Debugging
Code	High	Python	Use of sleep() in django
Code	High	Python	Incorrect Authorization for Critical Resources
Code	High	Python	Improper authorization
Code	Medium	Python	Sensitive Information Storage in Plaintext
Code	Medium	Python	Sensitive Information Transfer in Plaintext
Code	High	Python	Hardcoded Encryption Key
Code	High	Python	Missing login control
Code	Medium	Python	Exposure of system information
Code	Medium	Python	Exception Information Exposure
Code	Critical	Python	XQuery Injection
Code	High	Python	Reliance on DNS lookups in security decisions
Code	High	Python	Use of weak cryptographic algorithm
Code	Medium	Python	Weak Server Certificate
Code	High	Python	Hard coded sensitive information
Code	High	Python	Using Encryption Keys of Insufficient Size
Code	High	Python	Use of Insufficient Random Value
			Information Exposed from Cookies Saved in User''s

Code	Medium	Python	Hard Disk
Code	High	Python	Sensitive System Information Included in the Comment
Code	Medium	Python	One-way Hash Function Used Without Salt
Code	High	Python	Download of code without integrity check
Code	Medium	Python	Race Condition: Time Of Check and Time Of Use (TOCTOU)
Code	Medium	Python	Infinite Loop or Recursive Function
Code	Medium	Python	Error Message Information Exposed
Code	Medium	Python	Cannot Properly React to Errors
Code	Medium	Python	Improper exception handling
Code	Medium	Python	Null pointer dereference
Code	Medium	Python	Improper Resource Release
Code	High	Python	Deserialization of Untrusted Data
Code	Medium	Python	Debug Code Not Removed
Code	High	Python	Private Array Released from Public Method
Code	High	Python	Public Data Saved to Private Array
Code	High	Python	Allowing Critical Functions without Proper Authentication
Code	High	Python	Allowing Insecure Passwords
Code	Medium	Python	Cross-Site Scripting (XSS)
Code	High	SQL	SQL injection in iBatis
Code	High	SQL	SQL injection in myBatis
Code	High	Swift	Hardcoded Password
Code	High	Swift	Blank Password
Code	Medium	Swift	Weak hash algorithm
Code	Medium	Swift	Use of Cryptographic Algorithm in ECB Mode
Code	High	Swift	Hardcoded IP
Code	Medium	Swift	Transfer by GET
Code	High	Swift	Use of cookie with overly broad domain
Code	High	Swift	Use of Cookie with Overly Broad Path
Code	High	Swift	Storage of Sensitive Data in Persistent Cookie
Code	High	Swift	Empty HMAC Key
Code	High	Swift	Empty Salt
Code	High	Swift	Hardcoded Salt
Code	Medium	Swift	Insecure Cookie
Code	Medium	Swift	Reliance on weak frameworks in security decisions

Code	Low	Swift	Exposure of system information
Code	High	Swift	Weak SSL Protocol
Code	Low	Swift	Use of SMS Features
Code	Trivial	Swift	Absence of Auto-dialing Attack Prevention
Code	Low	Swift	HTTP header manipulation
Code	Medium	Swift	Key Derivation Function with Insecure Iteration Count
Code	Medium	Swift	Use of Null Salt
Code	High	Swift	SQL Injection
Code	High	Swift	Resource injection
Code	Medium	Swift	Cross-site scripting
Code	High	Swift	Command injection
Code	High	Swift	Sensitive Information Storage in Plaintext
Code	High	Swift	Hardcoded Encryption Key
Code	High	Swift	XPath Injection
Code	High	Swift	LDAP Injection
Code	Medium	Swift	Redirection to untrusted site
Code	High	Swift	Insufficient cryptographic key size
Code	Medium	Swift	Cleartext transmission of sensitive information
Code	High	Swift	Download of code without integrity check
Code	Medium	Swift	Empty catch block
Code	Medium	Swift	Catching overly broad exceptions
Code	High	Swift	Missing login control
Code	High	Swift	Weak password requirements
Code	Medium	Swift	Leftover debug code
Code	Medium	Swift	TOCTOU race condition
Code	Medium	Swift	HTTP Response Splitting
Code	High	Swift	Password in comment
Code	High	Swift	Format string injection
Code	High	Swift	Reliance on untrusted cookies in security decisions
Code	High	Swift	Missing authentication
Code	High	Swift	Improper authorization
Code	Medium	Swift	Reliance on DNS lookups in security decisions
Code	High	Swift	Improper random number generation
Code	High	Swift	Improper Reference of XML Entity
Code	Medium	Swift	Weak Server Certificate
Code	High	Swift	Deserialization of Untrusted Data

Code	High	TS	Using Strict Mode Group
Code	High	TS	Using Implicit Any Type
Code	Medium	TS	Clear Distinction of Null Type
Code	High	TS	Enabling alwaysStrict Mode
Code	Medium	Etc.	Enabled remote monitoring
Code	Medium	Etc.	Disabled bytecode verification
Code	Low	Etc.	User Account Information Hardcoded in JavaScript
Code	Medium	Etc.	Password Exposal in Comment
Code	Medium	Etc.	Hardcoded Password
Code	High	VB.Net	Command injection
Code	Medium	VB.Net	Empty catch block
Code	High	VB.Net	Hardcoded user name and password
Code	Medium	VB.Net	HTTP Response Splitting
Code	High	VB.Net	LDAP Injection
Code	Medium	VB.Net	Exposure of system information
Code	Medium	VB.Net	Redirection to untrusted site
Code	Low	VB.Net	Catching overly broad exceptions
Code	High	VB.Net	Path Traversal
Code	High	VB.Net	Reliance on DNS lookups in security decisions
Code	High	VB.Net	SQL Injection
Code	High	VB.Net	Unrestricted File Upload
Code	High	VB.Net	Weak cryptographic algorithm
Code	High	VB.Net	Insufficient cryptographic key size
Code	High	VB.Net	XPath Injection
Code	Critical	VB.Net	XQuery Injection
Code	High	VB.Net	Weak password requirements
Code	Medium	VB.Net	Cross-site scripting
Code	High	VB.Net	Use of Non-parameterized Query
Code	High	VB.Net	Inappropriate Signature
Code	Medium	VB.Net	Weak Server Certificate
Code	High	VBS	Overly Permissive CORS Policy
Code	High	VBS	Command injection
Code	High	VBS	Direct dynamic code evaluation
Code	High	VBS	Improper random number generation
Code	High	VBS	Exposure of system information
Code	Medium	VBS	Log injection

Code	Medium	VBS	Redirection to untrusted site
Code	High	VBS	Path Traversal
Code	Critical	VBS	Setting manipulation
Code	High	VBS	SQL Injection
Code	Medium	VBS	Insecure Reflection
Code	High	VBS	Weak cryptographic algorithm
Code	High	VBS	Weak hash
Code	High	VBS	Insufficient cryptographic key size
Code	Medium	VBS	Cross-site scripting
Code	Medium	VBS	Header manipulation
Code	High	VBS	Resource injection
Code	Medium	XML	Inappropriate Logging Level Setting
Code	Medium	XML	Duplicate Servlet Mapping
Code	Medium	XML	Duplicate Security Role
Code	Medium	XML	Excessive Servlet Mapping
Code	High	XML	Excessive Session Duration
Code	Medium	XML	Direct Access to JSP
Code	Medium	XML	Insufficient Session ID Length
Code	Medium	XML	Invalid Servlet Setting
Code	Medium	XML	Use of Non-existent Filter
Code	High	XML	Hardcoded Password
Code	High	XML	Debugging Information Exposal
Code	High	XML	Use of Default Error Page
Code	High	XML	Trace log exposure
Code	High	XML	Non-HttpOnly cookie
Code	Medium	XML	Struts: duplicate validation forms
Code	Medium	XML	Unused input validation of struts input
Open Source	Trivial	Common	Use of Components Licensed Under the 0BSD
Open Source	Low	Common	Use of Components Licensed Under the AAL
Open Source	High	Common	Use of Components Licensed Under the Abstyles
Open Source	High	Common	Use of Components Licensed Under the AdaCore-doc
Open Source	High	Common	Use of Components Licensed Under the Adobe-2006
Open Source	High	Common	Use of Components Licensed Under the Adobe-Glyph
Open Source	High	Common	Use of Components Licensed Under the ADSL
Open Source	Low	Common	Use of Components Licensed Under the AFL-1.1
Open Source	Low	Common	Use of Components Licensed Under the AFL-1.2

Open Source	Low	Common	Use of Components Licensed Under the AFL-2.0
Open Source	Low	Common	Use of Components Licensed Under the AFL-2.1
Open Source	Medium	Common	Use of Components Licensed Under the AFL-3.0
Open Source	High	Common	Use of Components Licensed Under the Afmparse
Open Source	Critical	Common	Use of Components Licensed Under the AGPL-1.0-only
Open Source	High	Common	Use of Components Licensed Under the AGPL-1.0-or-later
Open Source	Critical	Common	Use of Components Licensed Under the AGPL-3.0-only
Open Source	Medium	Common	Use of Components Licensed Under the AGPL-3.0-or-later
Open Source	Critical	Common	Use of Components Licensed Under the Aladdin
Open Source	High	Common	Use of Components Licensed Under the AMDPLPA
Open Source	Low	Common	Use of Components Licensed Under the AML
Open Source	Low	Common	Use of Components Licensed Under the AMPAS
Open Source	Low	Common	Use of Components Licensed Under the ANTLR-PD
Open Source	High	Common	Use of Components Licensed Under the ANTLR-PD-fallback
Open Source	High	Common	Use of Components Licensed Under the Apache-1.0
Open Source	Low	Common	Use of Components Licensed Under the Apache-1.1
Open Source	Low	Common	Use of Components Licensed Under the Apache-2.0
Open Source	Low	Common	Use of Components Licensed Under the APAFML
Open Source	Critical	Common	Use of Components Licensed Under the APL-1.0
Open Source	High	Common	Use of Components Licensed Under the App-s2p
Open Source	Critical	Common	Use of Components Licensed Under the APSL-1.0
Open Source	Critical	Common	Use of Components Licensed Under the APSL-1.1
Open Source	Critical	Common	Use of Components Licensed Under the APSL-1.2
Open Source	Critical	Common	Use of Components Licensed Under the APSL-2.0
Open Source	High	Common	Use of Components Licensed Under the Arphic-1999
Open Source	Critical	Common	Use of Components Licensed Under the Artistic-1.0
Open Source	High	Common	Use of Components Licensed Under the Artistic-1.0-cl8
Open Source	Critical	Common	Use of Components Licensed Under the Artistic-1.0-Perl
Open Source	Critical	Common	Use of Components Licensed Under the Artistic-2.0

Open Source	High	Common	Use of Components Licensed Under the Baekmuk
Open Source	High	Common	Use of Components Licensed Under the Bahyph
Open Source	High	Common	Use of Components Licensed Under the Barr
Open Source	Low	Common	Use of Components Licensed Under the Beerware
Open Source	High	Common	Use of Components Licensed Under the Bitstream-Charter
Open Source	High	Common	Use of Components Licensed Under the Bitstream-Vera
Open Source	High	Common	Use of Components Licensed Under the BitTorrent-1.0
Open Source	High	Common	Use of Components Licensed Under the BitTorrent-1.1
Open Source	Trivial	Common	Use of Components Licensed Under the blessing
Open Source	Low	Common	Use of Components Licensed Under the BlueOak-1.0.0
Open Source	High	Common	Use of Components Licensed Under the Borceux
Open Source	High	Common	Use of Components Licensed Under the Brian-Gladman-3-Clause
Open Source	Low	Common	Use of Components Licensed Under the BSD-1-Clause
Open Source	Low	Common	Use of Components Licensed Under the BSD-2-Clause
Open Source	Low	Common	Use of Components Licensed Under the BSD-2-Clause-FreeBSD
Open Source	High	Common	Use of Components Licensed Under the BSD-2-Clause-NetBSD
Open Source	Low	Common	Use of Components Licensed Under the BSD-2-Clause-Patent
Open Source	Low	Common	Use of Components Licensed Under the BSD-2-Clause-Views
Open Source	Low	Common	Use of Components Licensed Under the BSD-3-Clause
Open Source	Low	Common	Use of Components Licensed Under the BSD-3-Clause-Attribution
Open Source	Medium	Common	Use of Components Licensed Under the BSD-3-Clause-Clear
Open Source	Low	Common	Use of Components Licensed Under the BSD-3-Clause-LBNL
Open Source	High	Common	Use of Components Licensed Under the BSD-3-Clause-Modification

Open Source	High	Common	Use of Components Licensed Under the BSD-3-Clause-No-Military-License
Open Source	Low	Common	Use of Components Licensed Under the BSD-3-Clause-No-Nuclear-License
Open Source	Low	Common	Use of Components Licensed Under the BSD-3-Clause-No-Nuclear-License-2014
Open Source	High	Common	Use of Components Licensed Under the BSD-3-Clause-No-Nuclear-Warranty
Open Source	Low	Common	Use of Components Licensed Under the BSD-3-Clause-Open-MPI
Open Source	Low	Common	Use of Components Licensed Under the BSD-4-Clause
Open Source	High	Common	Use of Components Licensed Under the BSD-4-Clause-Shortened
Open Source	Low	Common	Use of Components Licensed Under the BSD-4-Clause-UC
Open Source	High	Common	Use of Components Licensed Under the BSD-4.3 RENO
Open Source	High	Common	Use of Components Licensed Under the BSD-4.3 TAHOE
Open Source	High	Common	Use of Components Licensed Under the BSD-Advertising-Acknowledgement
Open Source	High	Common	Use of Components Licensed Under the BSD-Attribution-HPND-disclaimer
Open Source	Medium	Common	Use of Components Licensed Under the BSD-Protection
Open Source	Low	Common	Use of Components Licensed Under the BSD-Source-Code
Open Source	Low	Common	Use of Components Licensed Under the BSL-1.0
Open Source	Medium	Common	Use of Components Licensed Under the BUSL-1.1
Open Source	High	Common	Use of Components Licensed Under the bzip2-1.0.5
Open Source	Low	Common	Use of Components Licensed Under the bzip2-1.0.6
Open Source	High	Common	Use of Components Licensed Under the C-UDA-1.0
Open Source	High	Common	Use of Components Licensed Under the CAL-1.0
Open Source	High	Common	Use of Components Licensed Under the CAL-1.0-Combined-Work-Exception
Open Source	High	Common	Use of Components Licensed Under the Caldera
Open Source	High	Common	Use of Components Licensed Under the CATOSL-1.1

Open Source	Low	Common	Use of Components Licensed Under the CC-BY-1.0
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-2.0
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-2.5
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-2.5-AU
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-3.0
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-3.0-AT
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-3.0-DE
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-3.0-IGO
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-3.0-NL
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-3.0-US
Open Source	Low	Common	Use of Components Licensed Under the CC-BY-4.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-1.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-2.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-2.5
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-3.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-3.0-DE
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-4.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-ND-1.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-ND-2.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-ND-2.5
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-ND-3.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-ND-3.0-DE

Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-ND-3.0-IGO
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-NC-ND-4.0
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-1.0
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-2.0
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-2.0-DE
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-2.0-FR
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-2.0-UK
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-2.5
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-3.0
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-3.0-DE
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-3.0-IGO
Open Source	Critical	Common	Use of Components Licensed Under the CC-BY-NC-SA-4.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-ND-1.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-ND-2.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-ND-2.5
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-ND-3.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-ND-3.0-DE
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-ND-4.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-1.0

Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-2.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-2.0-UK
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-2.1-JP
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-2.5
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-3.0
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-3.0-AT
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-3.0-DE
Open Source	Medium	Common	Use of Components Licensed Under the CC-BY-SA-4.0
Open Source	High	Common	Use of Components Licensed Under the CC-PDDC
Open Source	Trivial	Common	Use of Components Licensed Under the CC0-1.0
Open Source	Medium	Common	Use of Components Licensed Under the CDDL-1.0
Open Source	Medium	Common	Use of Components Licensed Under the CDDL-1.1
Open Source	High	Common	Use of Components Licensed Under the CDL-1.0
Open Source	Low	Common	Use of Components Licensed Under the CDLA-Permissive-1.0
Open Source	High	Common	Use of Components Licensed Under the CDLA-Permissive-2.0
Open Source	Medium	Common	Use of Components Licensed Under the CDLA-Sharing-1.0
Open Source	High	Common	Use of Components Licensed Under the CECILL-1.0
Open Source	High	Common	Use of Components Licensed Under the CECILL-1.1
Open Source	Critical	Common	Use of Components Licensed Under the CECILL-2.0
Open Source	Critical	Common	Use of Components Licensed Under the CECILL-2.1
Open Source	High	Common	Use of Components Licensed Under the CECILL-B
Open Source	Critical	Common	Use of Components Licensed Under the CECILL-C
Open Source	High	Common	Use of Components Licensed Under the CERN-OHL-1.1
Open Source	High	Common	Use of Components Licensed Under the CERN-OHL-1.2

Open Source	High	Common	Use of Components Licensed Under the CERN-OHL-P-2.0
Open Source	High	Common	Use of Components Licensed Under the CERN-OHL-S-2.0
Open Source	High	Common	Use of Components Licensed Under the CERN-OHL-W-2.0
Open Source	High	Common	Use of Components Licensed Under the CFITSIO
Open Source	High	Common	Use of Components Licensed Under the checkmk
Open Source	Critical	Common	Use of Components Licensed Under the ClArtistic
Open Source	High	Common	Use of Components Licensed Under the Clips
Open Source	High	Common	Use of Components Licensed Under the CMU-Mach
Open Source	High	Common	Use of Components Licensed Under the CNRI-Jython
Open Source	Low	Common	Use of Components Licensed Under the CNRI-Python
Open Source	High	Common	Use of Components Licensed Under the CNRI-Python-GPL-Compatible
Open Source	High	Common	Use of Components Licensed Under the COIL-1.0
Open Source	High	Common	Use of Components Licensed Under the Community-Spec-1.0
Open Source	High	Common	Use of Components Licensed Under the Condor-1.1
Open Source	Medium	Common	Use of Components Licensed Under the copyleft-next-0.3.0
Open Source	High	Common	Use of Components Licensed Under the copyleft-next-0.3.1
Open Source	High	Common	Use of Components Licensed Under the Cornell-Lossless-JPEG
Open Source	Critical	Common	Use of Components Licensed Under the CPAL-1.0
Open Source	Medium	Common	Use of Components Licensed Under the CPL-1.0
Open Source	Medium	Common	Use of Components Licensed Under the CPOL-1.02
Open Source	High	Common	Use of Components Licensed Under the Crossword
Open Source	High	Common	Use of Components Licensed Under the CrystalStacker
Open Source	High	Common	Use of Components Licensed Under the CUA-OPL-1.0
Open Source	High	Common	Use of Components Licensed Under the Cube
Open Source	Low	Common	Use of Components Licensed Under the curl
Open Source	High	Common	Use of Components Licensed Under the D-FSL-1.0
Open Source	High	Common	Use of Components Licensed Under the diffmark
Open Source	High	Common	Use of Components Licensed Under the DL-DE-BY-2.0
Open Source	Low	Common	Use of Components Licensed Under the DOC

Open Source	High	Common	Use of Components Licensed Under the Dotseqn
Open Source	High	Common	Use of Components Licensed Under the DRL-1.0
Open Source	High	Common	Use of Components Licensed Under the DSDP
Open Source	High	Common	Use of Components Licensed Under the dvipdfm
Open Source	Low	Common	Use of Components Licensed Under the ECL-1.0
Open Source	Low	Common	Use of Components Licensed Under the ECL-2.0
Open Source	High	Common	Use of Components Licensed Under the eCos-2.0
Open Source	Low	Common	Use of Components Licensed Under the EFL-1.0
Open Source	Low	Common	Use of Components Licensed Under the EFL-2.0
Open Source	High	Common	Use of Components Licensed Under the eGenix
Open Source	Medium	Common	Use of Components Licensed Under the Elastic-2.0
Open Source	Low	Common	Use of Components Licensed Under the Entessa
Open Source	High	Common	Use of Components Licensed Under the EPICS
Open Source	Medium	Common	Use of Components Licensed Under the EPL-1.0
Open Source	Medium	Common	Use of Components Licensed Under the EPL-2.0
Open Source	High	Common	Use of Components Licensed Under the ErlPL-1.1
Open Source	High	Common	Use of Components Licensed Under the etalab-2.0
Open Source	Low	Common	Use of Components Licensed Under the EUDatagrid
Open Source	High	Common	Use of Components Licensed Under the EUPL-1.0
Open Source	Critical	Common	Use of Components Licensed Under the EUPL-1.1
Open Source	Critical	Common	Use of Components Licensed Under the EUPL-1.2
Open Source	High	Common	Use of Components Licensed Under the Eurosym
Open Source	Low	Common	Use of Components Licensed Under the Fair
Open Source	High	Common	Use of Components Licensed Under the FDK-AAC
Open Source	Low	Common	Use of Components Licensed Under the Frameworx-1.0
Open Source	High	Common	Use of Components Licensed Under the FreeBSD-DOC
Open Source	High	Common	Use of Components Licensed Under the FreeImage
Open Source	Low	Common	Use of Components Licensed Under the FSFAP
Open Source	High	Common	Use of Components Licensed Under the FSFUL
Open Source	Low	Common	Use of Components Licensed Under the FSFULLR
Open Source	High	Common	Use of Components Licensed Under the FSFULLRWD
Open Source	Low	Common	Use of Components Licensed Under the FTL
Open Source	High	Common	Use of Components Licensed Under the GD
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.1-invariants-only

Open Source	High	Common	Use of Components Licensed Under the GFDL-1.1-invariants-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.1-no-invariants-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.1-no-invariants-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.1-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.1-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.2-invariants-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.2-invariants-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.2-no-invariants-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.2-no-invariants-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.2-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.2-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.3-invariants-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.3-invariants-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.3-no-invariants-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.3-no-invariants-or-later
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.3-only
Open Source	High	Common	Use of Components Licensed Under the GFDL-1.3-or-later
Open Source	High	Common	Use of Components Licensed Under the Giftware
Open Source	High	Common	Use of Components Licensed Under the GL2PS
Open Source	High	Common	Use of Components Licensed Under the Glide
Open Source	High	Common	Use of Components Licensed Under the Glulxe

Open Source	High	Common	Use of Components Licensed Under the GLWTPL
Open Source	Low	Common	Use of Components Licensed Under the gnuplot
Open Source	Medium	Common	Use of Components Licensed Under the GPL-1.0-only
Open Source	Medium	Common	Use of Components Licensed Under the GPL-1.0-or-later
Open Source	Medium	Common	Use of Components Licensed Under the GPL-2.0-only
Open Source	Medium	Common	Use of Components Licensed Under the GPL-2.0-or-later
Open Source	High	Common	Use of Components Licensed Under the GPL-2.0-with-autoconf-exception
Open Source	High	Common	Use of Components Licensed Under the GPL-2.0-with-bison-exception
Open Source	High	Common	Use of Components Licensed Under the GPL-2.0-with-classpath-exception
Open Source	High	Common	Use of Components Licensed Under the GPL-2.0-with-font-exception
Open Source	High	Common	Use of Components Licensed Under the GPL-2.0-with-GCC-exception
Open Source	Critical	Common	Use of Components Licensed Under the GPL-3.0-only
Open Source	Critical	Common	Use of Components Licensed Under the GPL-3.0-or-later
Open Source	High	Common	Use of Components Licensed Under the GPL-3.0-with-autoconf-exception
Open Source	High	Common	Use of Components Licensed Under the GPL-3.0-with-GCC-exception
Open Source	High	Common	Use of Components Licensed Under the Graphics-Gems
Open Source	High	Common	Use of Components Licensed Under the gSOAP-1.3b
Open Source	High	Common	Use of Components Licensed Under the HaskellReport
Open Source	High	Common	Use of Components Licensed Under the Hippocratic-2.1
Open Source	High	Common	Use of Components Licensed Under the HP-1986
Open Source	Low	Common	Use of Components Licensed Under the HPND
Open Source	High	Common	Use of Components Licensed Under the HPND-export-US
Open Source	High	Common	Use of Components Licensed Under the HPND-Markus-Kuhn

Open Source	Low	Common	Use of Components Licensed Under the HPND-sell-variant
Open Source	High	Common	Use of Components Licensed Under the HPND-sell-variant-MIT-disclaimer
Open Source	High	Common	Use of Components Licensed Under the HTMLTIDY
Open Source	High	Common	Use of Components Licensed Under the IBM-pibs
Open Source	Low	Common	Use of Components Licensed Under the ICU
Open Source	High	Common	Use of Components Licensed Under the IEC-Code-Components-EULA
Open Source	Low	Common	Use of Components Licensed Under the IJG
Open Source	High	Common	Use of Components Licensed Under the IJG-short
Open Source	High	Common	Use of Components Licensed Under the ImageMagick
Open Source	High	Common	Use of Components Licensed Under the iMatix
Open Source	High	Common	Use of Components Licensed Under the Imlib2
Open Source	High	Common	Use of Components Licensed Under the Info-ZIP
Open Source	Low	Common	Use of Components Licensed Under the Intel
Open Source	High	Common	Use of Components Licensed Under the Intel-ACPI
Open Source	Low	Common	Use of Components Licensed Under the Interbase-1.0
Open Source	Medium	Common	Use of Components Licensed Under the IPA
Open Source	Medium	Common	Use of Components Licensed Under the IPL-1.0
Open Source	Low	Common	Use of Components Licensed Under the ISC
Open Source	High	Common	Use of Components Licensed Under the Jam
Open Source	High	Common	Use of Components Licensed Under the JasPer-2.0
Open Source	High	Common	Use of Components Licensed Under the JPL-image
Open Source	High	Common	Use of Components Licensed Under the JPNIC
Open Source	Low	Common	Use of Components Licensed Under the JSON
Open Source	High	Common	Use of Components Licensed Under the Kazlib
Open Source	High	Common	Use of Components Licensed Under the KiCad-libraries-exception
Open Source	High	Common	Use of Components Licensed Under the Knuth-CTAN
Open Source	High	Common	Use of Components Licensed Under the LAL-1.2
Open Source	High	Common	Use of Components Licensed Under the LAL-1.3
Open Source	High	Common	Use of Components Licensed Under the Latex2e
Open Source	High	Common	Use of Components Licensed Under the Leptonica
Open Source	Medium	Common	Use of Components Licensed Under the LGPL-2.0-only

Open Source	Medium	Common	Use of Components Licensed Under the LGPL-2.0-or-later
Open Source	Medium	Common	Use of Components Licensed Under the LGPL-2.1-only
Open Source	Medium	Common	Use of Components Licensed Under the LGPL-2.1-or-later
Open Source	Critical	Common	Use of Components Licensed Under the LGPL-3.0-only
Open Source	Critical	Common	Use of Components Licensed Under the LGPL-3.0-or-later
Open Source	High	Common	Use of Components Licensed Under the LGPLLR
Open Source	Low	Common	Use of Components Licensed Under the Libpng
Open Source	Low	Common	Use of Components Licensed Under the libpng-2.0
Open Source	High	Common	Use of Components Licensed Under the libselinux-1.0
Open Source	Low	Common	Use of Components Licensed Under the libtiff
Open Source	High	Common	Use of Components Licensed Under the libutil-David-Nugent
Open Source	Low	Common	Use of Components Licensed Under the LiLiQ-P-1.1
Open Source	High	Common	Use of Components Licensed Under the LiLiQ-R-1.1
Open Source	High	Common	Use of Components Licensed Under the LiLiQ-Rplus-1.1
Open Source	High	Common	Use of Components Licensed Under the Linux-man-pages-copyleft
Open Source	High	Common	Use of Components Licensed Under the Linux-OpenIB
Open Source	High	Common	Use of Components Licensed Under the LOOP
Open Source	Low	Common	Use of Components Licensed Under the LPL-1.0
Open Source	Low	Common	Use of Components Licensed Under the LPL-1.02
Open Source	High	Common	Use of Components Licensed Under the LPPL-1.0
Open Source	High	Common	Use of Components Licensed Under the LPPL-1.1
Open Source	High	Common	Use of Components Licensed Under the LPPL-1.2
Open Source	High	Common	Use of Components Licensed Under the LPPL-1.3a
Open Source	High	Common	Use of Components Licensed Under the LPPL-1.3c
Open Source	High	Common	Use of Components Licensed Under the LZMA-SDK-9.11-to-9.20
Open Source	High	Common	Use of Components Licensed Under the LZMA-SDK-9.22
Open Source	High	Common	Use of Components Licensed Under the MakeIndex

Open Source	High	Common	Use of Components Licensed Under the Martin-Birgmeier
Open Source	High	Common	Use of Components Licensed Under the Minpack
Open Source	Low	Common	Use of Components Licensed Under the MirOS
Open Source	Low	Common	Use of Components Licensed Under the MIT
Open Source	Low	Common	Use of Components Licensed Under the MIT-0
Open Source	High	Common	Use of Components Licensed Under the MIT-advertising
Open Source	Low	Common	Use of Components Licensed Under the MIT-CMU
Open Source	High	Common	Use of Components Licensed Under the MIT-enna
Open Source	High	Common	Use of Components Licensed Under the MIT-feh
Open Source	High	Common	Use of Components Licensed Under the MIT-Modern-Variant
Open Source	High	Common	Use of Components Licensed Under the MIT-open-group
Open Source	High	Common	Use of Components Licensed Under the MIT-Wu
Open Source	High	Common	Use of Components Licensed Under the MITNFA
Open Source	High	Common	Use of Components Licensed Under the Motosoto
Open Source	High	Common	Use of Components Licensed Under the mpi-permissive
Open Source	High	Common	Use of Components Licensed Under the mpich2
Open Source	Medium	Common	Use of Components Licensed Under the MPL-1.0
Open Source	Medium	Common	Use of Components Licensed Under the MPL-1.1
Open Source	Medium	Common	Use of Components Licensed Under the MPL-2.0
Open Source	High	Common	Use of Components Licensed Under the MPL-2.0-no-copyleft-exception
Open Source	High	Common	Use of Components Licensed Under the mplus
Open Source	Medium	Common	Use of Components Licensed Under the MS-LPL
Open Source	Low	Common	Use of Components Licensed Under the MS-PL
Open Source	Medium	Common	Use of Components Licensed Under the MS-RL
Open Source	High	Common	Use of Components Licensed Under the MTLL
Open Source	High	Common	Use of Components Licensed Under the MulanPSL-1.0
Open Source	Low	Common	Use of Components Licensed Under the MulanPSL-2.0
Open Source	Low	Common	Use of Components Licensed Under the Multics
Open Source	High	Common	Use of Components Licensed Under the Mup

Open Source	High	Common	Use of Components Licensed Under the NAIST-2003
Open Source	High	Common	Use of Components Licensed Under the NASA-1.3
Open Source	Low	Common	Use of Components Licensed Under the Naumen
Open Source	High	Common	Use of Components Licensed Under the NBPL-1.0
Open Source	High	Common	Use of Components Licensed Under the NCGL-UK-2.0
Open Source	Low	Common	Use of Components Licensed Under the NCSA
Open Source	High	Common	Use of Components Licensed Under the Net-SNMP
Open Source	High	Common	Use of Components Licensed Under the NetCDF
Open Source	High	Common	Use of Components Licensed Under the Newsletr
Open Source	High	Common	Use of Components Licensed Under the NGPL
Open Source	High	Common	Use of Components Licensed Under the NICTA-1.0
Open Source	High	Common	Use of Components Licensed Under the NIST-PD
Open Source	High	Common	Use of Components Licensed Under the NIST-PD-fallback
Open Source	High	Common	Use of Components Licensed Under the NLOD-1.0
Open Source	High	Common	Use of Components Licensed Under the NLOD-2.0
Open Source	High	Common	Use of Components Licensed Under the NLPL
Open Source	Medium	Common	Use of Components Licensed Under the Nokia
Open Source	High	Common	Use of Components Licensed Under the NOSL
Open Source	High	Common	Use of Components Licensed Under the Noweb
Open Source	High	Common	Use of Components Licensed Under the NPL-1.0
Open Source	High	Common	Use of Components Licensed Under the NPL-1.1
Open Source	Critical	Common	Use of Components Licensed Under the NPOSL-3.0
Open Source	High	Common	Use of Components Licensed Under the NRL
Open Source	Low	Common	Use of Components Licensed Under the NTP
Open Source	High	Common	Use of Components Licensed Under the NTP-0
Open Source	High	Common	Use of Components Licensed Under the Nunit
Open Source	High	Common	Use of Components Licensed Under the O-UDA-1.0
Open Source	High	Common	Use of Components Licensed Under the OCCT-PL
Open Source	Low	Common	Use of Components Licensed Under the OCLC-2.0
Open Source	Medium	Common	Use of Components Licensed Under the ODbL-1.0
Open Source	Low	Common	Use of Components Licensed Under the ODC-By-1.0
Open Source	High	Common	Use of Components Licensed Under the OFFIS
Open Source	High	Common	Use of Components Licensed Under the OFL-1.0
Open Source	High	Common	Use of Components Licensed Under the OFL-1.0-no-RFN

Open Source	High	Common	Use of Components Licensed Under the OFL-1.0-RFN
Open Source	Medium	Common	Use of Components Licensed Under the OFL-1.1
Open Source	High	Common	Use of Components Licensed Under the OFL-1.1-no-RFN
Open Source	High	Common	Use of Components Licensed Under the OFL-1.1-RFN
Open Source	High	Common	Use of Components Licensed Under the OGC-1.0
Open Source	High	Common	Use of Components Licensed Under the OGDL-Taiwan-1.0
Open Source	High	Common	Use of Components Licensed Under the OGL-Canada-2.0
Open Source	High	Common	Use of Components Licensed Under the OGL-UK-1.0
Open Source	High	Common	Use of Components Licensed Under the OGL-UK-2.0
Open Source	High	Common	Use of Components Licensed Under the OGL-UK-3.0
Open Source	High	Common	Use of Components Licensed Under the OGTSL
Open Source	Medium	Common	Use of Components Licensed Under the OLDAP-1.1
Open Source	Medium	Common	Use of Components Licensed Under the OLDAP-1.2
Open Source	Medium	Common	Use of Components Licensed Under the OLDAP-1.3
Open Source	Medium	Common	Use of Components Licensed Under the OLDAP-1.4
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.0
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.0.1
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.1
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.2
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.2.1
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.2.2
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.3
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.4
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.5
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.6
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.7
Open Source	Low	Common	Use of Components Licensed Under the OLDAP-2.8
Open Source	High	Common	Use of Components Licensed Under the OML
Open Source	High	Common	Use of Components Licensed Under the OpenPBS-2.3
Open Source	Low	Common	Use of Components Licensed Under the OpenSSL
Open Source	High	Common	Use of Components Licensed Under the OPL-1.0
Open Source	High	Common	Use of Components Licensed Under the OPUBL-1.0
Open Source	High	Common	Use of Components Licensed Under the OSET-PL-2.1

Open Source	Critical	Common	Use of Components Licensed Under the OSL-1.0
Open Source	Critical	Common	Use of Components Licensed Under the OSL-1.1
Open Source	Critical	Common	Use of Components Licensed Under the OSL-2.0
Open Source	Critical	Common	Use of Components Licensed Under the OSL-2.1
Open Source	Critical	Common	Use of Components Licensed Under the OSL-3.0
Open Source	High	Common	Use of Components Licensed Under the Parity-6.0.0
Open Source	High	Common	Use of Components Licensed Under the Parity-7.0.0
Open Source	High	Common	Use of Components Licensed Under the PDDL-1.0
Open Source	High	Common	Use of Components Licensed Under the PHP-3.0
Open Source	High	Common	Use of Components Licensed Under the PHP-3.01
Open Source	High	Common	Use of Components Licensed Under the Plexus
Open Source	High	Common	Use of Components Licensed Under the PolyForm-Noncommercial-1.0.0
Open Source	High	Common	Use of Components Licensed Under the PolyForm-Small-Business-1.0.0
Open Source	Low	Common	Use of Components Licensed Under the PostgreSQL
Open Source	Low	Common	Use of Components Licensed Under the PSF-2.0
Open Source	High	Common	Use of Components Licensed Under the psfrag
Open Source	High	Common	Use of Components Licensed Under the psutils
Open Source	Low	Common	Use of Components Licensed Under the Python-2.0
Open Source	High	Common	Use of Components Licensed Under the Python-2.0.1
Open Source	High	Common	Use of Components Licensed Under the Qhull
Open Source	Medium	Common	Use of Components Licensed Under the QPL-1.0
Open Source	High	Common	Use of Components Licensed Under the QPL-1.0-INRIA-2004
Open Source	High	Common	Use of Components Licensed Under the Rdisc
Open Source	High	Common	Use of Components Licensed Under the RHeCos-1.1
Open Source	Critical	Common	Use of Components Licensed Under the RPL-1.1
Open Source	Critical	Common	Use of Components Licensed Under the RPL-1.5
Open Source	Critical	Common	Use of Components Licensed Under the RPSL-1.0
Open Source	Low	Common	Use of Components Licensed Under the RSA-MD
Open Source	High	Common	Use of Components Licensed Under the RSCPL
Open Source	Critical	Common	Use of Components Licensed Under the Ruby
Open Source	High	Common	Use of Components Licensed Under the SAX-PD
Open Source	High	Common	Use of Components Licensed Under the Saxpath
Open Source	High	Common	Use of Components Licensed Under the SCEA

Open Source	High	Common	Use of Components Licensed Under the Sendmail
Open Source	High	Common	Use of Components Licensed Under the Sendmail-8.23
Open Source	Low	Common	Use of Components Licensed Under the SGI-B-1.0
Open Source	Low	Common	Use of Components Licensed Under the SGI-B-1.1
Open Source	Low	Common	Use of Components Licensed Under the SGI-B-2.0
Open Source	High	Common	Use of Components Licensed Under the SHL-0.5
Open Source	High	Common	Use of Components Licensed Under the SHL-0.51
Open Source	High	Common	Use of Components Licensed Under the SimPL-2.0
Open Source	High	Common	Use of Components Licensed Under the SISSL
Open Source	High	Common	Use of Components Licensed Under the SISSL-1.2
Open Source	High	Common	Use of Components Licensed Under the Sleepycat
Open Source	High	Common	Use of Components Licensed Under the SMLNJ
Open Source	High	Common	Use of Components Licensed Under the SMPPL
Open Source	High	Common	Use of Components Licensed Under the SNIA
Open Source	High	Common	Use of Components Licensed Under the snprintf
Open Source	High	Common	Use of Components Licensed Under the Spencer-86
Open Source	High	Common	Use of Components Licensed Under the Spencer-94
Open Source	High	Common	Use of Components Licensed Under the Spencer-99
Open Source	High	Common	Use of Components Licensed Under the SPL-1.0
Open Source	High	Common	Use of Components Licensed Under the SSH-OpenSSH
Open Source	High	Common	Use of Components Licensed Under the SSH-short
Open Source	Medium	Common	Use of Components Licensed Under the SSPL-1.0
Open Source	High	Common	Use of Components Licensed Under the SugarCRM-1.1.3
Open Source	High	Common	Use of Components Licensed Under the SunPro
Open Source	High	Common	Use of Components Licensed Under the SWL
Open Source	High	Common	Use of Components Licensed Under the Symlinks
Open Source	High	Common	Use of Components Licensed Under the TAPR-OHL-1.0
Open Source	High	Common	Use of Components Licensed Under the TCL
Open Source	High	Common	Use of Components Licensed Under the TCP-wrappers
Open Source	High	Common	Use of Components Licensed Under the TMate
Open Source	High	Common	Use of Components Licensed Under the TORQUE-1.1

Open Source	High	Common	Use of Components Licensed Under the TOSL
Open Source	High	Common	Use of Components Licensed Under the TPDL
Open Source	High	Common	Use of Components Licensed Under the TPL-1.0
Open Source	High	Common	Use of Components Licensed Under the TTWL
Open Source	High	Common	Use of Components Licensed Under the TU-Berlin-1.0
Open Source	High	Common	Use of Components Licensed Under the TU-Berlin-2.0
Open Source	High	Common	Use of Components Licensed Under the UCAR
Open Source	High	Common	Use of Components Licensed Under the UCL-1.0
Open Source	Low	Common	Use of Components Licensed Under the Unicode-DFS-2015
Open Source	Low	Common	Use of Components Licensed Under the Unicode-DFS-2016
Open Source	Low	Common	Use of Components Licensed Under the Unicode-TOU
Open Source	Trivial	Common	Use of Components Licensed Under the Unlicense
Open Source	Low	Common	Use of Components Licensed Under the UPL-1.0
Open Source	Critical	Common	Use of Components Licensed Under the Vim
Open Source	High	Common	Use of Components Licensed Under the VOSTROM
Open Source	High	Common	Use of Components Licensed Under the VSL-1.0
Open Source	Low	Common	Use of Components Licensed Under the W3C
Open Source	Low	Common	Use of Components Licensed Under the W3C-19980720
Open Source	Low	Common	Use of Components Licensed Under the W3C-20150513
Open Source	High	Common	Use of Components Licensed Under the w3m
Open Source	High	Common	Use of Components Licensed Under the Watcom-1.0
Open Source	High	Common	Use of Components Licensed Under the Wsuipa
Open Source	Trivial	Common	Use of Components Licensed Under the WTFPL
Open Source	High	Common	Use of Components Licensed Under the wxWindows
Open Source	Low	Common	Use of Components Licensed Under the X11
Open Source	High	Common	Use of Components Licensed Under the X11-distribute-modifications-variant
Open Source	High	Common	Use of Components Licensed Under the Xerox
Open Source	Low	Common	Use of Components Licensed Under the XFree86-1.1
Open Source	Low	Common	Use of Components Licensed Under the xinetd
Open Source	High	Common	Use of Components Licensed Under the xlock
Open Source	High	Common	Use of Components Licensed Under the Xnet

Open Source	Low	Common	Use of Components Licensed Under the xpp
Open Source	High	Common	Use of Components Licensed Under the XSkat
Open Source	High	Common	Use of Components Licensed Under the YPL-1.0
Open Source	High	Common	Use of Components Licensed Under the YPL-1.1
Open Source	High	Common	Use of Components Licensed Under the ZCL-1.1
Open Source	High	Common	Use of Components Licensed Under the Zed
Open Source	High	Common	Use of Components Licensed Under the Zend-2.0
Open Source	High	Common	Use of Components Licensed Under the Zimbra-1.3
Open Source	High	Common	Use of Components Licensed Under the Zimbra-1.4
Open Source	Low	Common	Use of Components Licensed Under the Zlib
Open Source	Low	Common	Use of Components Licensed Under the zlib-acknowledgement
Open Source	High	Common	Use of Components Licensed Under the ZPL-1.1
Open Source	Low	Common	Use of Components Licensed Under the ZPL-2.0
Open Source	Low	Common	Use of Components Licensed Under the ZPL-2.1
Open Source	High	Common	Using the 3D-Slicer-1.0 Licensed Component
Open Source	High	Common	Using AMD-newlib Licensed Components
Open Source	High	Common	Using the BSD-2-Clause-first-lines license component
Open Source	High	Common	Using Catharon licensed components
Open Source	High	Common	Using Gutmann Licensed Components
Open Source	High	Common	Using HPND-Intel Licensed Components
Open Source	High	Common	Use the HPND-UC-export-US license component
Open Source	High	Common	Use the HPND-export-US-acknowledgement license component
Open Source	High	Common	Using the HPND-export2-US licensed component
Open Source	High	Common	Using the HPND-merchantability-variant license component
Open Source	High	Common	Use the HPND-sell-variant-MIT-disclaimer-rev license component
Open Source	High	Common	Using MIT-Khronos old license components
Open Source	High	Common	Using NCBI-PD licensed components
Open Source	High	Common	Using NCL license components
Open Source	High	Common	Using OAR licensed components
Open Source	High	Common	Using PPL licensed components
Open Source	High	Common	Using the Sun-PPP-2000 Licensed Component
Open Source	High	Common	Using any-OSI licensed components

Open Source	High	Common	Using CVE-TOU licensed components
Open Source	High	Common	Using pkgconf licensed components
Open Source	High	Common	Using threeparttable licensed components
Open Source	High	Common	Using xzoom licensed components
Open Source	High	Common	Using the Adobe-Display-PostScript Licensed Component
Open Source	High	Common	Use Adobe-Utopia licensed components
Open Source	High	Common	Using the AML-glslang Licensed Component
Open Source	High	Common	Using the ASWF-Digital-Assets-1.0 Licensed Component
Open Source	High	Common	Using the ASWF-Digital-Assets-1.1 Licensed Component
Open Source	High	Common	Using the bcrypt-Solar-Designer licensed component
Open Source	High	Common	Using the Boehm-GC Licensed Component
Open Source	High	Common	Using the Brian-Gladman-2-Clause License Component
Open Source	High	Common	Using the BSD-2-Clause-Darwin License Component
Open Source	High	Common	Using BSD-3-Clause-acpica Licensed Components
Open Source	High	Common	Using the BSD-3-Clause-flex License Component
Open Source	Medium	Common	Use CC-BY-SA-3.0-IGO licensed components
Open Source	High	Common	Using BSD-3-Clause-HP Licensed Components
Open Source	High	Common	Using the BSD-3-Clause-Sun License Component
Open Source	High	Common	Using BSD-Inferno-Nettverk Licensed Components
Open Source	High	Common	Using the BSD-Source-beginning-file license component
Open Source	High	Common	Using BSD-Systemics Licensed Components
Open Source	High	Common	Using BSD-Systemics-W3Works Licensed Components
Open Source	High	Common	Using Caldera-no-preamble licensed components
Open Source	High	Common	Use CC-BY-3.0-AU licensed components
Open Source	High	Common	Using the check-cvs license component
Open Source	High	Common	Using the CMU-Mach-nodoc licensed component
Open Source	High	Common	Using Cronyx licensed components
Open Source	High	Common	Using the DEC-3-Clause License Component
Open Source	High	Common	Using DL-DE-ZERO-2.0 licensed components
Open Source	High	Common	Using DocBook-Schema licensed components
Open Source	High	Common	Using the DocBook-XML Licensing Component

Open Source	High	Common	Using the DRL-1.1 License Component
Open Source	High	Common	Using DTOA license components
Open Source	High	Common	Using FBM licensed components
Open Source	High	Common	Using Ferguson-Twofish licensed components
Open Source	High	Common	Using the FSFAP-no-warranty-disclaimer license component
Open Source	High	Common	Using Furuseth Licensed Components
Open Source	High	Common	Using FWLW licensed components
Open Source	High	Common	Using GCR-docs licensed components
Open Source	High	Common	Using the LPD-document license component
Open Source	High	Common	Using the gtkbook licensed component
Open Source	High	Common	Using HDPARM licensed components
Open Source	High	Common	Using HIDAPI Licensed Components
Open Source	High	Common	Using the HPND-doc-sell licensing component
Open Source	High	Common	Using the HP-1989 License Component
Open Source	High	Common	Using HPND-DEC licensed components
Open Source	High	Common	Using HPND-doc licensed components
Open Source	High	Common	Use the HPND-export-US-modify license component
Open Source	High	Common	Using HPND-Fenneberg-Livingston licensed components
Open Source	High	Common	Using HPND-INRIA-IMAG licensed components
Open Source	High	Common	Using HPND-Kevlin-Henney Licensed Components
Open Source	High	Common	Using the HPND-MIT-disclaimer license component
Open Source	High	Common	Using HPND-Netrek licensed components
Open Source	High	Common	Using HPND-Pbmplus licensed components
Open Source	High	Common	Use the HPND-sell-MIT-disclaimer-xserver license component
Open Source	High	Common	Using the HPND-sell-regexpr licensing component
Open Source	High	Common	Using HPND-UC licensed components
Open Source	High	Common	Using Kastrup licensed components
Open Source	High	Common	Using Inner-Net-2.0 licensed components
Open Source	High	Common	Using ISC-Veillard Licensed Components
Open Source	High	Common	Using the Latex2e-translated-notice license component
Open Source	High	Common	Using the Linux-man-pages-1-para licensed component

Open Source	High	Common	Using the Linux-man-pages-copyleft-2-para licensed component
Open Source	High	Common	Using the Linux-man-pages-copyleft-var license component
Open Source	High	Common	Using LSOF license components
Open Source	High	Common	Using the Lucida-Bitmap-Fonts Licensed Component
Open Source	High	Common	Using the Mackerras-3-Clause License Component
Open Source	High	Common	Using the Mackerras-3-Clause-acknowledgment License Component
Open Source	High	Common	Using magaz licensed components
Open Source	High	Common	Using mailprio licensed components
Open Source	High	Common	Using the McPhee-slideshow licensed component
Open Source	High	Common	Using metamail licensed components
Open Source	High	Common	Using MIT-Festival licensed components
Open Source	High	Common	Using MIT-testregex licensed components
Open Source	High	Common	Using MMIXware licensed components
Open Source	High	Common	Using MPEG-SSG Licensed Components
Open Source	High	Common	Using NIST-Software licensed components
Open Source	High	Common	Using the OLFL-1.3 License Component
Open Source	High	Common	Using OpenSSL-standalone licensed components
Open Source	High	Common	Using OpenVision Licensed Components
Open Source	High	Common	Using OPL-UK-3.0 licensed components
Open Source	High	Common	Using the PADL license component
Open Source	High	Common	Using Pixar Licensed Components
Open Source	High	Common	Using pnmstitch licensed components
Open Source	High	Common	Using the python-ldap licensed component
Open Source	High	Common	Using the RADVD License Component
Open Source	High	Common	Using Ruby-empty licensed components
Open Source	High	Common	Using SAX-PD-2.0 Licensed Components
Open Source	High	Common	Using SchemeReport licensed components
Open Source	High	Common	Using Sun-PPP licensed components
Open Source	High	Common	Using SGI-OpenGL Licensed Components
Open Source	High	Common	Using SGP4 licensed components
Open Source	High	Common	Using the Ubuntu-font-1.0 licensed component
Open Source	High	Common	Using SL licensed components
Open Source	High	Common	Using softSurfer licensed components

Open Source	High	Common	Using Soundex Licensed Components
Open Source	High	Common	Using the ssh-keyscan license component
Open Source	High	Common	Using the SSLeay-standalone licensed component
Open Source	High	Common	Using the swrule license component
Open Source	High	Common	Using TermReadKey licensed components
Open Source	High	Common	Using TGPPL-1.0 licensed components
Open Source	High	Common	Using TTYP0 license components
Open Source	High	Common	Using ULEM licensed components
Open Source	High	Common	Using the UMich-Merit license component
Open Source	High	Common	Using Unicode-3.0 Licensed Components
Open Source	High	Common	Using UnixCrypt licensed components
Open Source	High	Common	Using the URT-RLE License Component
Open Source	High	Common	Using Widget-Workshop Licensed Components
Open Source	High	Common	Using X11-swapped licensed components
Open Source	High	Common	Using the Xdebug-1.03 Licensed Component
Open Source	High	Common	Using Xfig Licensed Components
Open Source	High	Common	xkeyboard-config - Using the Zinoviev licensed component
Open Source	High	Common	Using Zeeff Licensed Components
Open Source	Critical	Common	Use of Vulnerable Component