

CWE Standards Model Summary for C / C++

The LDRA tool suite ® is developed and certified to BS EN ISO 9001:2015, TÜV SÜD and SGS-TÜV Saar.

This information is applicable to version 10.2.0 of the LDRA tool suite®.
It is correct as of 11th September 2023.
© Copyright 2023 LDRA Ltd. All rights reserved

Compliance is measured against "Common Weakness Enumeration"
A Community-Developed Dictionary of Software Weakness Types. Version 3.4"
23rd September 2019
Copyright © The MITRE Corporation
Further information is available at <http://cwe.mitre.org>

Classification	Enhanced Enforcement	Fully Implemented	Partially Implemented	Not yet Implemented	Not statically Checkable	Total
Rule	16	44	156	0	0	216
Total	16	44	156	0	0	216

CWE Standards Model Compliance for C / C++

Rule	Classification	Rule Description	LDRA Standard	LDRA Standard Description
14	Rule	Compiler Removal of Code to Clear Buffers	105 D	DU anomaly dead code, var value is unused on all paths. Modifiers : 391 = 0 (Default)
		This rule is a parent rule for 37, 74, 77, 78, 88, 119, 120, 121, 122, 123, 124, 125, 126, 127, 129, 134, 138, 160, 170, 190, 415, 416, 464, 466, 606, 786, 787, 788, 805, 806, 823, 824, 825, 1173 which has mappings to: 42 D: Local pointer returned in function result. 43 D: Divide by zero found. 51 D: Attempt to read from freed memory. 69 D: UR anomaly, variable used before assignment. 77 D: Local structure returned in function result. 85 D: Filename not verified before fopen. 86 D: User input not checked before use. 109 D: Tainted argument to formatted i/o function. 112 D: Free called twice on same variable. 127 D: Local or member denominator not checked before use.		

20	Rule	Improper Input Validation	131 D: Global denominator not checked within this procedure. 137 D: Parameter used as denominator not checked before use. 139 D: Incorrect array index validation. 140 D: Copy source parameter not checked before use. 141 D: Heap-based array index not checked before use. 44 S: Use of banned function, type or variable. 47 S: Array bound exceeded. 71 S: Pointer assignment to wider scope. 377 S: Null character used within a string. 401 S: Use of sizeof on an array parameter. 403 S: Negative (or potentially negative) shift. 447 S: Loop termination not a simple constant test. 484 S: Attempt to use already freed object. 486 S: Incorrect number of formats in output function. 487 S: Insufficient space allocated. 488 S: Value outside range of underlying type. 489 S: Insufficient space for operation. 493 S: Numeric overflow. 503 S: Function returns local resources. 564 S: Reference assignment to wider scope. 567 S: Pointer arithmetic is not on array. 584 S: Remainder of % op could be negative. 588 S: Use of system function. 600 S: Argument of strlen is unterminated. 602 S: strtok may change the parse string. 629 S: Divide by zero found. 687 S: Loop counter not checked for negative value. 688 S: Memory allocation exceeds 2 power *** bytes. 692 S: Array index is negative. 1 X: Declaration types do not match across a system. 64 X: Array bound exceeded at call. 66 X: Insufficient array space at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 70 X: Array has insufficient space. 71 X: Insufficient space for copy. 79 X: Size mismatch in memcpy/memset. 80 X: Divide by zero found.		
22	Rule	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	This rule is a parent rule for 36, 37, 38, 39 which has mappings to		
36	Rule	Absolute Path Traversal	This rule is a parent rule for 37, 38, 39 which has mappings to:		
37	Rule	Path Traversal: '/absolute/pathname/here'		85 D	Filename not verified before fopen.
38	Rule	Path Traversal: '\\absolute\\pathname\\here'		85 D	Filename not verified before fopen.
39	Rule	Path Traversal: 'C:dirname'		85 D	Filename not verified before fopen.
41	Rule	Improper Resolution of Path Equivalence		85 D	Filename not verified before fopen.
59	Rule	Improper Link Resolution Before File Access ('Link Following')		85 D	Filename not verified before fopen.
74	Rule	Improper Neutralization of Special Elements in Output Used by a Downstream Component ('Injection')	This rule is a parent rule for 37, 77, 78, 88, 138, 160, 170, 464 which has mappings to: 85 D: Filename not verified before fopen. 44 S: Use of banned function, type or variable. 377 S: Null character used within a string. 588 S: Use of system function. 600 S: Argument of strlen is unterminated. 602 S: strtok may change the parse string.		
77	Rule	Improper Neutralization of Special Elements used in a Command ('Command Injection')			
78	Rule	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	The analysis highlights the use of system, and the windows _exec and __spawn family of commands. These are considered undesirable.	44 S	Use of banned function, type or variable.
				588 S	Use of system function.

88	Rule	Improper Neutralization of Argument Delimiters in a Command ('Argument Injection')		588 S	Use of system function.
116	Rule	Improper Encoding or Escaping of Output	This rule is a parent rule for 37, 74, 77, 78, 88, 138, 160, 170, 464 which has mappings to: 85 D: Filename not verified before fopen. 44 S: Use of banned function, type or variable. 377 S: Null character used within a string. 588 S: Use of system function. 600 S: Argument of strlen is unterminated. 602 S: strtok may change the parse string.		
118	Rule	Incorrect Access of Indexable Resource ('Range Error')	This rule is a parent rule for 119, 120, 121, 122, 123, 124, 125, 126, 127, 415, 416, 466, 786, 787, 788, 805, 806, 823, 824, 825 which has mappings to: 42 D: Local pointer returned in function result. 51 D: Attempt to read from freed memory. 69 D: UR anomaly, variable used before assignment. 77 D: Local structure returned in function result. 86 D: User input not checked before use. 109 D: Tainted argument to formatted i/o function. 112 D: Free called twice on same variable. 139 D: Incorrect array index validation. 140 D: Copy source parameter not checked before use. 141 D: Heap-based array index not checked before use. 44 S: Use of banned function, type or variable. 47 S: Array bound exceeded. 71 S: Pointer assignment to wider scope. 401 S: Use of sizeof on an array parameter. 484 S: Attempt to use already freed object. 489 S: Insufficient space for operation. 503 S: Function returns local resources. 564 S: Reference assignment to wider scope. 567 S: Pointer arithmetic is not on array. 584 S: Remainder of % op could be negative. 687 S: Loop counter not checked for negative value. 688 S: Memory allocation exceeds 2 power *** bytes. 692 S: Array index is negative. 64 X: Array bound exceeded at call. 66 X: Insufficient array space at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 70 X: Array has insufficient space. 71 X: Insufficient space for copy. 79 X: Size mismatch in memcpy/memset.		
119	Rule	Improper Restriction of Operations within the Bounds of a Memory Buffer	This rule is a parent rule for 120, 121, 122, 123, 124, 125, 126, 127, 415, 416, 466, 786, 787, 788, 805, 806, 823, 824, 825 which has mappings to: 42 D: Local pointer returned in function result. 51 D: Attempt to read from freed memory. 69 D: UR anomaly, variable used before assignment. 77 D: Local structure returned in function result. 86 D: User input not checked before use. 112 D: Free called twice on same variable. 140 D: Copy source parameter not checked before use. 141 D: Heap-based array index not checked before use. 44 S: Use of banned function, type or variable. 47 S: Array bound exceeded. 71 S: Pointer assignment to wider scope. 484 S: Attempt to use already freed object. 489 S: Insufficient space for operation. 503 S: Function returns local resources. 564 S: Reference assignment to wider scope. 567 S: Pointer arithmetic is not on array. 584 S: Remainder of % op could be negative. 687 S: Loop counter not checked for negative value. 688 S: Memory allocation exceeds 2 power *** bytes. 692 S: Array index is negative. 64 X: Array bound exceeded at call. 66 X: Insufficient array space at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 70 X: Array has insufficient space. 71 X: Insufficient space for copy. 79 X: Size mismatch in memcpy/memset. The following standards are used to implement this rule	109 D	Tainted argument to formatted i/o function.
				139 D	Incorrect array index validation.
				401 S	Use of sizeof on an array parameter.
120	Rule	Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')		109 D	Tainted argument to formatted i/o function.
				44 S	Use of banned function, type or variable.
				489 S	Insufficient space for operation.
				66 X	Insufficient array space at call.

				70 X	Array has insufficient space.
				71 X	Insufficient space for copy.
121	Rule	Stack-based Buffer Overflow		86 D	User input not checked before use.
				140 D	Copy source parameter not checked before use.
				47 S	Array bound exceeded.
				489 S	Insufficient space for operation.
				584 S	Remainder of % op could be negative.
				687 S	Loop counter not checked for negative value.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				66 X	Insufficient array space at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
				70 X	Array has insufficient space.
				71 X	Insufficient space for copy.
				79 X	Size mismatch in memcpy/memset.
122	Rule	Heap-based Buffer Overflow		86 D	User input not checked before use.
				140 D	Copy source parameter not checked before use.
				141 D	Heap-based array index not checked before use.
				47 S	Array bound exceeded.
				489 S	Insufficient space for operation.
				584 S	Remainder of % op could be negative.
				688 S	Memory allocation exceeds 2 power *** bytes.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				66 X	Insufficient array space at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
				70 X	Array has insufficient space.
				71 X	Insufficient space for copy.
				79 X	Size mismatch in memcpy/memset.
123	Rule	Write-what-where Condition		86 D	User input not checked before use.
124	Rule	Buffer Underwrite ('Buffer Underflow')		47 S	Array bound exceeded.
				584 S	Remainder of % op could be negative.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.

125	Rule	Out-of-bounds Read	This rule is a parent rule for 126, 127 which has mappings to: 47 S: Array bound exceeded. 584 S: Remainder of % op could be negative. 692 S: Array index is negative. 64 X: Array bound exceeded at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use.		
126	Rule	Buffer Over-read		47 S	Array bound exceeded.
				584 S	Remainder of % op could be negative.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
127	Rule	Buffer Under-read		47 S	Array bound exceeded.
				584 S	Remainder of % op could be negative.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
128	Rule	Wrap-around Error		493 S	Numeric overflow.
				488 S	Value outside range of underlying type.
129	Rule	Improper Validation of Array Index		43 D	Divide by zero found.
				127 D	Local or member denominator not checked before use.
				131 D	Global denominator not checked within this procedure.
				137 D	Parameter used as denominator not checked before use.
				139 D	Incorrect array index validation.
				141 D	Heap-based array index not checked before use.
				47 S	Array bound exceeded.
				401 S	Use of sizeof on an array parameter.
				584 S	Remainder of % op could be negative.
				629 S	Divide by zero found.
				692 S	Array index is negative.
				1 X	Declaration types do not match across a system.
				64 X	Array bound exceeded at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
				80 X	Divide by zero found.

131	Rule	Incorrect Calculation of Buffer Size		115 D	Copy length parameter not checked before use.
				487 S	Insufficient space allocated.
				638 S	Memory allocation non-conformant with type.
134	Rule	Use of Externally-Controlled Format String		86 D	User input not checked before use.
				486 S	Incorrect number of formats in output function.
138	Rule	Improper Neutralization of Special Elements	This rule is a parent rule for 37, 160, 170, 464 which has mappings to: 85 D: Filename not verified before fopen. 377 S: Null character used within a string. 600 S: Argument of strlen is unterminated. 602 S: strtok may change the parse string.		
160	Rule	Improper Neutralization of Leading Special Elements	This rule is a parent rule for 37 which has mappings to: 85 D:Filename not verified before fopen.		
170	Rule	Improper Null Termination		377 S	Null character used within a string.
				600 S	Argument of strlen is unterminated.
172	Rule	Encoding Error	This rule is a parent rule for 176 which has mappings to: 44 S: Use of banned function, type or variable.		
176	Rule	Improper Handling of Unicode Encoding	The analysis highlights the use of the Windows functions MultiByteToWideChar, WideCharToMultiByte, UnicodeToBytes and BytesToUnicode.	44 S	Use of banned function, type or variable.
187	Rule	Partial String Comparison	The analysis highlights the use of strstr and strncmp, which are considered undesirable.	44 S	Use of banned function, type or variable.
188	Rule	Reliance on Data/Memory Layout	The analysis highlights the use of pointer arithmetic on all objects that are not declared as arrays.	567 S	Pointer arithmetic is not on array. Modifiers :436 = 0 (Default)
				618 S	Use of memcmp between structures.
190	Rule	Integer Overflow or Wraparound		43 D	Divide by zero found.
				403 S	Negative (or potentially negative) shift.
				487 S	Insufficient space allocated.
				488 S	Value outside range of underlying type. Modifiers :535 = 0 (Default)
				493 S	Numeric overflow.
				629 S	Divide by zero found.
				80 X	Divide by zero found.
191	Rule	Integer Underflow (Wrap or Wraparound)		493 S	Numeric overflow.
				488 S	Value outside range of underlying type. Modifiers :535 = 0 (Default)
193	Rule	Off-by-one Error		47 S	Array bound exceeded.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
194	Rule	Unexpected Sign Extension		446 S	Narrower int conversion without cast. Modifiers :374 = 0 (Default)

195	Rule	Signed to Unsigned Conversion Error		101 S	Function return type inconsistent. Modifiers :374 = 0 (Default) 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 430 = 0 (Default) 431 = 1 453 = 0 (Default) 470 = 0 (Default)
				107 S	Type mismatch in ternary expression. Modifiers : 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 470 = 0 (Default) 525 = 0 (Default)
				331 S	Literal value requires a U suffix. Modifiers :309 = 0 (Default) 358 = 1 516 = 0 (Default)
				434 S	Signed/unsigned conversion without cast. Modifiers :374 = 0 (Default) 358 = 1 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 430 = 0 (Default) 470 = 0 (Default)
				442 S	Signed integral type cast to unsigned. Modifiers : 470 = 0 (Default) 507 = 1 529 = 1
196	Rule	Unsigned to Signed Conversion Error		101 S	Function return type inconsistent. Modifiers : 374 = 0 (Default) 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 430 = 0 (Default) 431 = 1 453 = 0 (Default) 470 = 0 (Default)
				107 S	Type mismatch in ternary expression. Modifiers : 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 470 = 0 (Default) 525 = 0 (Default)
				434 S	Signed/unsigned conversion without cast. Modifiers :374 = 0 (Default) 358 = 1 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 430 = 0 (Default) 470 = 0 (Default)
				443 S	Unsigned integral type cast to signed. Modifiers :508 = 1 529 = 1
				52 S	Unsigned expression negated.
				101 S	Function return type inconsistent.. Modifiers :374 = 0 (Default) 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 430 = 0 (Default) 431 = 1 453 = 0 (Default) 470 = 0 (Default)
				107 S	Type mismatch in ternary expression. Modifiers : 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 470 = 0 (Default) 525 = 0 (Default)

197	Rule	Numeric Truncation Error		433 S	Type conversion without cast. Modifiers :374 = 0 (Default) 391 = 0 (Default) 431 = 1 428 = c : 0 (Default), c++ : 1 (Default) 470 = 0 (Default) 386 = 1 (Default) 549 = 0 (Default)
				434 S	Signed/unsigned conversion without cast. Modifiers :374 = 0 (Default) 358 = 1 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 430 = 0 (Default) 470 = 0 (Default)
				435 S	Float/integer conversion without cast. Modifiers :374 = 0 (Default) 432 = 0 (Default) 453 = 0 (Default)
				442 S	Signed integral type cast to unsigned. Modifiers :470 = 0 (Default) 507 = 1 529 = 1
				443 S	Unsigned integral type cast to signed. Modifiers :508 = 1 529 = 1
				446 S	Narrower int conversion without cast. Modifiers :374 = 0 (Default)
				458 S	Implicit conversion: actual to formal param (MR).
				493 S	Numeric overflow.
				488 S	Value outside range of underlying type. Modifiers :535 = 0 (Default)
216	Rule	Containment Errors (Container Errors)	This rule is a parent rule for 493, 500 which has mappings to: 38 S: Use of static class member.		
228	Rule	Improper Handling of Syntactically Invalid Structure	This rule is a parent rule for 229, 232, 233, 234, 235 which has mappings to: 45 D: Pointer not checked for null before use. 21 S: Number of parameters does not match.		
229	Rule	Improper Handling of Values	This rule is a parent rule for 232 which has mappings to: 45 D: Pointer not checked for null before use.		
232	Rule	Improper Handling of Undefined Values		45 D	Pointer not checked for null before use.
233	Rule	Improper Handling of Parameters	This rule is a parent rule for 234, 235 which has mappings to: 21 S: Number of parameters does not match.		
234	Rule	Failure to Handle Missing Parameter		21 S	Number of parameters does not match.
235	Rule	Improper Handling of Extra Parameters		21 S	Number of parameters does not match.
242	Rule	Use of Inherently Dangerous Function	The analysis highlights the use of gets, which is considered undesirable. Other functions can be added by the user.	44 S	Use of banned function, type or variable.
248	Rule	Uncaught Exception		527 S	No master exception handler.
				80 D	Potentially unused function-modified value.
				91 D	Function return value potentially unused.
				124 D	Var set by std lib func return not checked before use.

252	Rule	Unchecked Return Value		130 D	Global set by std lib func return not checked before use.
				382 S	(void) missing for discarded return value. Modifiers : 383 = 1
				593 S	Use fseek() rather than rewind().
				594 S	Use setvbuf() rather than setbuf().
253	Rule	Incorrect Check of Function Return Value		531 S	Literal zero used in pointer context.
269	Rule	Improper Privilege Management	This rule is a parent rule for 271, 273 which has mappings to: 80 D: Potentially unused function-modified value. 91 D: Function return value potentially unused. 124 D: Var set by std lib func return not checked before use. 130 D: Global set by std lib func return not checked before use. 382 S: (void) missing for discarded return value.		
271	Rule	Privilege Dropping / Lowering Errors	This rule is a parent rule for 273 which has mappings to: 80 D: Potentially unused function-modified value. 91 D: Function return value potentially unused. 124 D: Var set by std lib func return not checked before use. 130 D: Global set by std lib func return not checked before use. 382 S: (void) missing for discarded return value.		
273	Rule	Improper Check for Dropped Privileges		80 D	Potentially unused function-modified value.
				91 D	Function return value potentially unused.
				124 D	Var set by std lib func return not checked before use.
				130 D	Global set by std lib func return not checked before use.
				382 S	(void) missing for discarded return value. Modifiers :383 = 1
284	Rule	Improper Access Control	This rule is a parent rule for 269, 271, 273 which has mappings to: 80 D: Potentially unused function-modified value. 91 D: Function return value potentially unused. 124 D: Var set by std lib func return not checked before use. 130 D: Global set by std lib func return not checked before use. 382 S: (void) missing for discarded return value		
327	Rule	Use of a Broken or Risky Cryptographic Algorithm	The analysis looks for evidence of character manipulation, but can not determine whether it is a good or bad algorithm. In addition 633 S highlights the use of certain parameters to CryptDeriveKey.	329 S	Operation not appropriate to plain char. Modifiers :191 = 1 331 = 0 (Default) 391 = 0 (Default)
				633 S	Use of a broken or risky cryptography algorithm.
330	Rule	Use of Insufficiently Random Values	This rule is a parent rule for 338, 344, 587 which has mappings to: 44 S: Use of banned function, type or variable. 440 S: Cast from integral type to pointer.		
338	Rule	Use of Cryptographically Weak Pseudo-Random Number Generator (PRNG)	The analysis highlights the use of rand, which is considered undesirable. The following standards are used to implement this rule	44 S	Use of banned function, type or variable.
344	Rule	Use of Invariant Value in Dynamically Changing Context	This rule is a parent rule for 587 which has mappings to: 440 S: Cast from integral type to pointer.		

362	Rule	Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	This rule is a parent rule for 364, 365, 367, 479, 828 which has mappings to: 88 D: Illegal use of longjmp in signal handler. 89 D: Illegal use of raise in signal handler. 97 D: Signal called from within signal handler. 44 S: Use of banned function, type or variable. 592 S: Use of filename based functions.	75 D	Attempt to open file pointer more than once.
				90 D	Unguarded global used in threaded function.
364	Rule	Signal Handler Race Condition	This rule is a parent rule for 479, 828 which has mappings to: 88 D: Illegal use of longjmp in signal handler. 89 D: Illegal use of raise in signal handler. 97 D: Signal called from within signal handler.		
365	Rule	Race Condition in Switch	The analysis highlights all uses of time, not just those in switch choosers. The following standards are used to implement this rule	44 S	Use of banned function, type or variable.
367	Rule	Time-of-check Time-of-use (TOCTOU) Race Condition		592 S	Use of filename based functions.
369	Rule	Divide By Zero		43 D	Divide by zero found.
				127 D	Local or member denominator not checked before use.
				131 D	Global denominator not checked within this procedure.
				137 D	Parameter used as denominator not checked before use.
				248 S	Divide by zero in preprocessor directive.
				629 S	Divide by zero found.
				80 X	Divide by zero found.
390	Rule	Detection of Error Condition Without Action		634 S	Empty block found. Modifiers :422 = 1
391	Rule	Unchecked Error Condition		80 D	Potentially unused function-modified value.
				91 D	Function return value potentially unused.
				124 D	Var set by std lib func return not checked before use.
				130 D	Global set by std lib func return not checked before use.
400	Rule	Uncontrolled Resource Consumption		49 D	File pointer not closed on exit.
				50 D	Memory not freed after last reference.
				75 D	Attempt to open file pointer more than once.
				81 D	Attempt to remove an open file.
401	Rule	Missing Release of Memory after Effective Lifetime		50 D	Memory not freed after last reference.
402	Rule	Transmission of Private Resources into a New Sphere ('Resource Leak')	This rule is a parent rule for 403 which has mappings to: 49 D: File pointer not closed on exit.		
403	Rule	Exposure of File Descriptor to Unintended Control Sphere ('File Descriptor Leak')		49 D	File pointer not closed on exit.

404	Rule	Improper Resource Shutdown or Release	This rule is a parent rule for 401, 459, 460, 590, 762, 763, 772, 775, 1091 which has mappings to: 50 D: Memory not freed after last reference. 56 D: Throw found with no catch in scope. 71 D: No matching catch for throw in called function. 75 D: Attempt to open file pointer more than once. 81 D: Attempt to remove an open file. 125 D: free called on variable with no allocated space. 407 S: free used on string. 483 S: Freed parameter is not heap item. 485 S: Array deletion without []. 644 S: realloc ptr does not originate from allocation function.	49 D	File pointer not closed on exit.
405	Rule	Asymmetric Resource Consumption (Amplification)	This rule is a parent rule for 1042, 1176 which has mappings to: 38 S: Use of static class member.		
415	Rule	Double Free		112 D	Free called twice on same variable.
				484 S	Attempt to use already freed object. Modifiers :350 = 1
416	Rule	Use After Free		51 D	Attempt to read from freed memory.
				484 S	Attempt to use already freed object. Modifiers :350 = 1
435	Rule	Improper Interaction Between Multiple Correctly-Behaving Entities	This rule is a parent rule for 14, 188, 733, 1037, 1038 which has mappings to: 76 D: Procedure is not called or referenced in code analysed. 105 D: DU anomaly dead code, var value is unused on all paths. 1 J: Unreachable Code found. 23 S: Procedure is not explicitly called in code analysed. 35 S: Static procedure is not explicitly called in code analysed. 139 S: Construct leads to infeasible code. 140 S: Infeasible loop condition found. 567 S: Pointer arithmetic is not on array. 618 S: Use of memcmp between structures.		
456	Rule	Missing Initialization of a Variable	Standards mapped to CWE-457 catch other consequences of missing initialisations.	57 D	Global not initialised at declaration.
457	Rule	Use of Uninitialized Variable		53 D	Attempt to use uninitialised pointer.
				54 D	Unsafe use of function pointer variable.
				69 D	UR anomaly, variable used before assignment.
				631 S	Declaration not reachable.
				652 S	Object created by malloc used before initialisation.
459	Rule	Incomplete Cleanup	This rule is a parent rule for 460 which has mappings to: 56 D: Throw found with no catch in scope. 71 D: No matching catch for throw in called function.		
460	Rule	Improper Cleanup on Thrown Exception		56 D	Throw found with no catch in scope.
				71 D	No matching catch for throw in called function.
464	Rule	Addition of Data Structure Sentinel		377 S	Null character used within a string.
				602 S	strtok may change the parse string.
466	Rule	Return of Pointer Value Outside of Expected Range		47 S	Array bound exceeded.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.

467	Rule	Use of sizeof() on a Pointer Type		401 S	Use of sizeof on an array parameter.
				577 S	Sizeof argument is a pointer.
468	Rule	Incorrect Pointer Scaling		53 D	Attempt to use uninitialised pointer.
				54 D	Unsafe use of function pointer variable.
				438 S	Pointer subtraction not addressing one array.
				567 S	Pointer arithmetic is not on array. Modifiers :436 = 0 (Default)
				576 S	Function pointer is of wrong type.
469	Rule	Use of Pointer Subtraction to Determine Size	438 S is used to indicate subtraction between arrays and strings, but will not be issued of additional pointer indirections are present. 567 S highlights all uses of pointer arithmetic which includes pointer subtraction.	438 S	Pointer subtraction not addressing one array.
				567 S	Pointer arithmetic is not on array. Modifiers :436 = 0 (Default)
475	Rule	Undefined Behavior for Input to API	The analysis highlights the use of memcpy, which is considered undesirable.	44 S	Use of banned function, type or variable.
476	Rule	NULL Pointer Dereference		45 D	Pointer not checked for null before use.
				123 D	File pointer not checked for null before use.
				128 D	Global pointer not checked within this procedure.
				129 D	Global file pointer not checked within this procedure.
				135 D	Pointer assigned to NULL may be dereferenced.
				136 D	Global pointer assigned to NULL may be dereferenced.
477	Rule	Use of Obsolete Function	The analysis highlights a limited number of obsolete functions such as getpwd. Details on adding other functions can be found in the documentation for 44 S.	44 S	Use of banned function, type or variable.
478	Rule	Missing Default Case in Switch Statement		48 S	No default case in switch statement. Modifiers :252 = 0 (Default) 233 = 0 (Default)
479	Rule	Signal Handler Use of a Non-reentrant Function		88 D	Illegal use of longjmp in signal handler.
				89 D	Illegal use of raise in signal handler.
				97 D	Signal called from within signal handler.
480	Rule	Use of Incorrect Operator	This rule is a parent rule for 481, 482, 597 which has mappings to: 57 S: Statement with no side effect. 132 S: Assignment operator in boolean expression. 583 S: Comparison of pointer and string literal.	99 S	Function use is not a call. Modifiers :498 = 0 (Default)
				136 S	Bit operator with boolean operand.
481	Rule	Assigning instead of Comparing		132 S	Assignment operator in boolean expression.
482	Rule	Comparing instead of Assigning		57 S	Statement with no side effect. Modifiers :458 = 0 (Default)
483	Rule	Incorrect Block Delimitation	The analysis will produce violations for all blocks without brackets. This is stricter than 483, which allows an unbracketed statement if the layout suggests that only one statement was intended.	11 S	No brackets to loop body (added by Testbed).
				12 S	No brackets to then/else (added by Testbed).
				428 S	No {} for switch (added by Testbed).
484	Rule	Omitted Break Statement in Switch		62 S	Switch case not terminated with break. Modifiers :283 = 0 (Default) 384 = c++ : 0 (Default) 551 = 0 (Default)
493	Rule	Critical Public Variable Without Final Modifier	This rule is a parent rule for 500 which has mappings to: 38 S: Use of static class member.		

500	Rule	Public Static Field Not Marked Final		38 S	Use of static class member. Modifiers :423 = 1
547	Rule	Use of Hard-coded, Security-relevant Constants	The analysis highlights the use of all numeric literals values (except 0, 1 and those literals used in #define and as enumeration constants).	201 S	Use of numeric literal in expression. Modifiers :217 = 1 163 = 1 (Default) 237 = 0 (Default) 207 = 0 (Default)
558	Rule	Use of getlogin() in Multithreaded Application	The analysis highlights all uses of getlogin. The must then manually check whether the call occurs in a multi-threaded environment	44 S	Use of banned function, type or variable.
560	Rule	Use of umask() with chmod-style Argument	The analysis highlights all uses of umask. The must then manually check whether the call occurs with a chmod-style argument.	44 S	Use of banned function, type or variable.
561	Rule	Dead Code		65 D	Void function has no side effects.
				76 D	Procedure is not called or referenced in code analysed.
				1 J	Unreachable Code found.
				23 S	Procedure is not explicitly called in code analysed.
				35 S	Static procedure is not explicitly called in code analysed.
				139 S	Construct leads to infeasible code.
				140 S	Infeasible loop condition found.
				631 S	Declaration not reachable.
				1 U	Inter-file recursion found.
562	Rule	Return of Stack Variable Address		42 D	Local pointer returned in function result.
				77 D	Local structure returned in function result.
				71 S	Pointer assignment to wider scope. Modifiers :222 = 1
				503 S	Function returns local resources.
				564 S	Reference assignment to wider scope.
563	Rule	Assignment to Variable without Use		94 D	Named variable declared but not used in code.
				105 D	DU anomaly dead code, var value is unused on all paths.
				151 D	DD data flow anomaly found.
570	Rule	Expression is Always False		1 J	Unreachable Code found.
				139 S	Construct leads to infeasible code.
				140 S	Infeasible loop condition found.
571	Rule	Expression is Always True		1 J	Unreachable Code found.
				139 S	Construct leads to infeasible code.
				140 S	Infeasible loop condition found.

573	Rule	Improper Following of Specification by Caller	This rule is a parent rule for 253, 415, 475, 560, 628, 675, 685, 686, 687, 688 which has mappings to: 41 D: Procedure call has no prototype declared. 75 D: Attempt to open file pointer more than once. 83 D: Potentially repeated call to ungetc. 112 D: Free called twice on same variable. 20 S: Parameter not declared explicitly. 21 S: Number of parameters does not match. 41 S: Ellipsis used in procedure parameter list. 44 S: Use of banned function, type or variable. 135 S: Parameter list is KR. 148 S: No return type for function/procedure. 170 S: Procedure call has no prototype and no defn. 326 S: Declaration is missing type. 458 S: Implicit conversion: actual to formal param (MR). 484 S: Attempt to use already freed object. 486 S: Incorrect number of formats in output function. 496 S: Function call with no prior declaration. 531 S: Literal zero used in pointer context. 576 S: Function pointer is of wrong type. 589 S: Format is not appropriate type.		
587	Rule	Assignment of a Fixed Address to a Pointer		440 S	Cast from integral type to pointer. Modifiers :397 = 0 (Default) 548 = 0 (Default)
588	Rule	Attempt to Access Child of a Non-structure Pointer		440 S	Cast from integral type to pointer. Modifiers :397 = 0 (Default) 548 = 0 (Default)
				540 S	Cast from pointer to void to pointer. Modifiers :381 = 0 (Default)
590	Rule	Free of Memory not on the Heap	The analysis warns of the use of free on a non-heap object. The document of 125 D describes how this may be extended to cover other deallocation functions.	125 D	free called on variable with no allocated space.
				407 S	free used on string.
				483 S	Freed parameter is not heap item.
				644 S	realloc ptr does not originate from allocation function.
595	Rule	Comparison of Object References Instead of Object Contents	This rule is a parent rule for 597 which has mappings to: 583 S: Comparison of pointer and string literal.		
597	Rule	Use of Wrong Operator in String Comparison		583 S	Comparison of pointer and string literal.
606	Rule	Unchecked Input for Loop Condition	The analysis highlights loops that do not have a simple terminating loop condition	447 S	Loop termination not a simple constant test.
628	Rule	Function Call with Incorrectly Specified Arguments	This rule is a parent rule for 560, 685, 686, 687, 688 which has mappings to: 21 S: Number of parameters does not match. 44 S: Use of banned function, type or variable. 458 S: Implicit conversion: actual to formal param (MR). 486 S: Incorrect number of formats in output function. 576 S: Function pointer is of wrong type. 589 S: Format is not appropriate type.	41 D	Procedure call has no prototype declared.
				20 S	Parameter not declared explicitly.
				41 S	Ellipsis used in procedure parameter list. Modifiers :522 = 1
				135 S	Parameter list is KR.
				148 S	No return type for function/procedure.
				170 S	Procedure call has no prototype and no defn.
				326 S	Declaration is missing type.
				496 S	Function call with no prior declaration.
662	Rule	Improper Synchronization		87 D	Illegal shared object in signal handler.
663	Rule	Use of a Non-reentrant Function in a Concurrent Context	This rule is a parent rule for 479, 558 which has mappings to: 88 D: Illegal use of longjmp in signal handler. 89 D: Illegal use of raise in signal handler. 97 D: Signal called from within signal handler. 44 S: Use of banned function, type or variable.		

664	Rule	Improper Control of a Resource Through its Lifetime	This rule is a parent rule for 22, 36, 37, 38, 39, 41, 59, 134, 194, 195, 196, 197, 216, 269, 271, 273, 284, 400, 401, 402, 403, 404, 405, 415, 416, 456, 457, 459, 460, 479, 493, 500, 558, 588, 590, 662, 663, 665, 666, 668, 672, 681, 704, 706, 762, 763, 767, 772, 775, 825, 843, 908, 909, 910, 1042, 1052, 1058, 1091, 1176, 1187 which has mappings to: 42 D: Local pointer returned in function result. 48 D: Attempt to write to unopened file. 49 D: File pointer not closed on exit. 50 D: Memory not freed after last reference. 51 D: Attempt to read from freed memory. 53 D: Attempt to use uninitialised pointer. 54 D: Unsafe use of function pointer variable. 56 D: Throw found with no catch in scope. 57 D: Global not initialised at declaration. 69 D: UR anomaly, variable used before assignment. 71 D: No matching catch for throw in called function. 75 D: Attempt to open file pointer more than once. 77 D: Local structure returned in function result. 80 D: Potentially unused function-modified value. 81 D: Attempt to remove an open file. 85 D: Filename not verified before fopen. 86 D: User input not checked before use. 87 D: Illegal shared object in signal handler. 88 D: Illegal use of longjmp in signal handler. 89 D: Illegal use of raise in signal handler. 91 D: Function return value potentially unused. 97 D: Signal called from within signal handler. 112 D: Free called twice on same variable. 113 D: File closed more than once. 124 D: Var set by std lib func return not checked before use. 125 D: free called on variable with no allocated space. 126 D: File closed that was never opened. 130 D: Global set by std lib func return not checked before use. 38 S: Use of static class member. 44 S: Use of banned function, type or variable. 52 S: Unsigned expression negated. 71 S: Pointer assignment to wider scope. 101 S: Function return type inconsistent. 107 S: Type mismatch in ternary expression. 127 S: Array has no bounds specified. 201 S: Use of numeric literal in expression. 203 S: Cast on a constant value. 331 S: Literal value requires a U suffix. 344 S: Cast on volatile value. 382 S: (void) missing for discarded return value. 392 S: Class data accessible thru non const member. 403 S: Negative (or potentially negative) shift. 404 S: Array initialisation has too many items. 407 S: free used on string. 411 S: Inappropriate value assigned to enum. 433 S: Type conversion without cast. 434 S: Signed/unsigned conversion without cast. 435 S: Float/integer conversion without cast. 439 S: Cast from pointer to integral type. 440 S: Cast from integral type to pointer. 441 S: Float cast to non-float. 442 S: Signed integral type cast to unsigned. 443 S: Unsigned integral type cast to signed. 444 S: Integral type cast to non-integral. 446 S: Narrower int conversion without cast. 456 S: Implicit float widening for function return (MR). 458 S: Implicit conversion: actual to formal param (MR). 483 S: Freed parameter is not heap item. 484 S: Attempt to use already freed object. 485 S: Array deletion without []. 486 S: Incorrect number of formats in output function. 493 S: Numeric overflow. 503 S: Function returns local resources. 540 S: Cast from pointer to void to pointer. 545 S: Assignment of overlapping storage. 564 S: Reference assignment to wider scope. 631 S: Declaration not reachable. 644 S: realloc ptr does not originate from allocation function. 652 S: Object created by malloc used before initialisation. 671 S: Class data accessible thru non const handle. 691 S: Initialisation of pointer with initialiser list.	406 S	Use of ++ or -- on RHS of && or operator.
				408 S	Volatile variable accessed on RHS of && or .
665	Rule	Improper Initialization	This rule is a parent rule for 456, 457, 909, 1052, 1187 which has mappings to: 53 D: Attempt to use uninitialised pointer. 54 D: Unsafe use of function pointer variable. 57 D: Global not initialised at declaration. 201 S: Use of numeric literal in expression. 631 S: Declaration not reachable. 652 S: Object created by malloc used before initialisation.	69 D	UR anomaly, variable used before assignment.
				127 S	Array has no bounds specified. Modifiers :285 = 0 (Default) 286 = 1 426 = 1
				404 S	Array initialisation has too many items. Modifiers :399 = 1
				691 S	Initialisation of pointer with initialiser list.

666	Rule	Operation on Resource in Wrong Phase of Lifetime	This rule is a parent rule for 415, 416, 672, 825, 910 which has mappings to: 42 D: Local pointer returned in function result. 48 D: Attempt to write to unopened file. 51 D: Attempt to read from freed memory. 77 D: Local structure returned in function result. 112 D: Free called twice on same variable. 113 D: File closed more than once. 126 D: File closed that was never opened. 71 S: Pointer assignment to wider scope. 484 S: Attempt to use already freed object. 503 S: Function returns local resources. 564 S: Reference assignment to wider scope.		
668	Rule	Exposure of Resource to Wrong Sphere	This rule is a parent rule for 22, 36, 37, 38, 39, 134, 402, 403, 493, 500, 767 which has mappings to: 49 D: File pointer not closed on exit. 85 D: Filename not verified before fopen. 86 D: User input not checked before use. 38 S: Use of static class member. 392 S: Class data accessible thru non const member. 486 S: Incorrect number of formats in output function. 671 S: Class data accessible thru non const handle.		
670	Rule	Always-Incorrect Control Flow Implementation	This rule is a parent rule for 480, 481, 482, 483, 484, 597, 783 which has mappings to: 11 S: No brackets to loop body (added by Testbed). 12 S: No brackets to then/else (added by Testbed). 49 S: Logical conjunctions need brackets. 57 S: Statement with no side effect. 62 S: Switch case not terminated with break. 99 S: Function use is not a call. 132 S: Assignment operator in boolean expression. 136 S: Bit operator with boolean operand. 361 S: Expression needs brackets. 428 S: No {} for switch (added by Testbed). 583 S: Comparison of pointer and string literal.		
672	Rule	Operation on a Resource after Expiration or Release	This rule is a parent rule for 415, 416, 825, 910 which has mappings to: 42 D: Local pointer returned in function result. 48 D: Attempt to write to unopened file. 51 D: Attempt to read from freed memory. 77 D: Local structure returned in function result. 112 D: Free called twice on same variable. 113 D: File closed more than once. 126 D: File closed that was never opened. 71 S: Pointer assignment to wider scope. 484 S: Attempt to use already freed object. 503 S: Function returns local resources. 564 S: Reference assignment to wider scope.		
674	Rule	Uncontrolled Recursion		6 D	Recursion in procedure calls found.
				1 U	Inter-file recursion found.
675	Rule	Duplicate Operations on Resource	This rule is a parent rule for 415 which has mappings to: 112 D: Free called twice on same variable. 484 S: Attempt to use already freed object.	75 D	Attempt to open file pointer more than once.
				83 D	Potentially repeated call to ungetc.
676	Rule	Use of Potentially Dangerous Function	The analysis highlights the C/C++ standard library functions listed in Microsoft's banned.h. Details on adding/removing functions can be found in the documentation for 44 S.	44 S	Use of banned function, type or variable.
				592 S	Use of filename based functions.
				52 S	Unsigned expression negated.
				101 S	Function return type inconsistent. Modifiers :374 = 0 (Default) 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 430 = 0 (Default) 431 = 1 453 = 0 (Default) 470 = 0 (Default)
				107 S	Type mismatch in ternary expression. Modifiers : 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 470 = 0 (Default) 525 = 0 (Default)
				403 S	Negative (or potentially negative) shift.

681	Rule	Incorrect Conversion between Numeric Types	The analysis does not highlight the casting to narrower type. This rule is a parent rule for 194, 195, 196, 197 which has mappings to: 331 S: Literal value requires a U suffix.	411 S	Inappropriate value assigned to enum.
				433 S	Type conversion without cast. Modifiers :374 = 0 (Default) 391 = 0 (Default) 431 = 1 428 = c : 0 (Default), c++ : 1 (Default) 470 = 0 (Default) 386 = 1 (Default) 549 = 0 (Default)
				434 S	Signed/unsigned conversion without cast. Modifiers : 374 = 0 (Default) 358 = 1 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 430 = 0 (Default) 470 = 0 (Default)
				435 S	Float/integer conversion without cast. Modifiers : 374 = 0 (Default) 432 = 0 (Default) 453 = 0 (Default)
				439 S	Cast from pointer to integral type. Modifiers :548 = 0 (Default)
				440 S	Cast from integral type to pointer. Modifiers :397 = 0 (Default) 548 = 0 (Default)
				441 S	Float cast to non-float. Modifiers :502 = 1 503 = 0 (Default) 504 = 1 505 = 0 (Default) 506 = 1 529 = 1
				442 S	Signed integral type cast to unsigned. Modifiers :470 = 0 (Default) 507 = 1 529 = 1
				443 S	Unsigned integral type cast to signed. Modifiers :508 = 1 529 = 1
				444 S	Integral type cast to non-integral. Modifiers :509 = 1 510 = 0 (Default) 511 = 1 512 = 0 (Default) 513 = 1 529 = 1
				446 S	Narrower int conversion without cast. Modifiers :374 = 0 (Default)
				456 S	Implicit float widening for function return (MR). Modifiers :191 = 1
				458 S	Implicit conversion: actual to formal param (MR).
				493 S	Numeric overflow.
				488 S	Value outside range of underlying type. Modifiers : 535 = 0 (Default)
				39 S	Unsuitable type for loop variable.
				50 S	Use of shift operator on signed type.
				54 S	Sizeof operator with side effects.

682	Rule	Incorrect Calculation	This rule is a parent rule for 128, 131, 190, 191, 193, 369, 467, 468, 469 which has mappings to: 43 D: Divide by zero found. 53 D: Attempt to use uninitialised pointer. 54 D: Unsafe use of function pointer variable. 115 D: Copy length parameter not checked before use. 127 D: Local or member denominator not checked before use. 131 D: Global denominator not checked within this procedure. 137 D: Parameter used as denominator not checked before use. 47 S: Array bound exceeded. 248 S: Divide by zero in preprocessor directive. 401 S: Use of sizeof on an array parameter. 403 S: Negative (or potentially negative) shift. 438 S: Pointer subtraction not addressing one array. 487 S: Insufficient space allocated. 488 S: Value outside range of underlying type. 493 S: Numeric overflow. 567 S: Pointer arithmetic is not on array. 576 S: Function pointer is of wrong type. 577 S: Sizeof argument is a pointer. 629 S: Divide by zero found. 638 S: Memory allocation non-conformant with type. 692 S: Array index is negative. 64 X: Array bound exceeded at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 80 X: Divide by zero found.	56 S	Equality comparison of floating point. Modifiers :223 = 1
				329 S	Operation not appropriate to plain char. Modifiers :191 = 1 331 = 0 (Default) 391 = 0 (Default)
				332 S	Widening cast on complex integer expression. Modifiers :529 = 1
				389 S	Bool value incremented/decremented.
				397 S	Array initialisation has insufficient items. Modifiers :415 = 0 (Default) 445 = 0 (Default) 450 = 0 (Default) 455 = 0 (Default)
				406 S	Use of ++ or -- on RHS of && or operator.
				408 S	Volatile variable accessed on RHS of && or .
				452 S	No cast for widening complex int expression (MR). Modifiers :191 = 1 529 = 1
				479 S	Right shift loses all bits.
				491 S	No cast for widening int parameter.
				506 S	Use of boolean with relational operator.
				584 S	Remainder of % op could be negative.
				585 S	Bitwise and arith operations on same data.
				653 S	Apparent side effects in _Generic or _Alignof.
685	Rule	Function Call With Incorrect Number of Arguments		21 S	Number of parameters does not match.
				486 S	Incorrect number of formats in output function.
686	Rule	Function Call With Incorrect Argument Type		458 S	Implicit conversion: actual to formal param (MR).
				486 S	Incorrect number of formats in output function.
				576 S	Function pointer is of wrong type.
				589 S	Format is not appropriate type.
687	Rule	Function Call With Incorrectly Specified Argument Value	This rule is a parent rule for 560 which has mappings to: 44 S: Use of banned function, type or variable.		
688	Rule	Function Call With Incorrect Variable or Reference as Argument		486 S	Incorrect number of formats in output function.
				576 S	Function pointer is of wrong type.
				589 S	Format is not appropriate type.

691	Rule	Insufficient Control Flow Management	This rule is a parent rule for 248, 362, 364, 365, 367, 479, 480, 481, 482, 483, 484, 558, 597, 662, 663, 670, 674, 705, 768, 783, 828, 834, 835, 1058 which has mappings to: 5 C: Procedure contains infinite loop. 6 D: Recursion in procedure calls found. 28 D: Potentially infinite loop found. 35 D: Expression has side effects. 75 D: Attempt to open file pointer more than once. 87 D: Illegal shared object in signal handler. 88 D: Illegal use of longjmp in signal handler. 89 D: Illegal use of raise in signal handler. 90 D: Unguarded global used in threaded function. 97 D: Signal called from within signal handler. 1 Q: Call has execution order dependant side effects. 11 S: No brackets to loop body (added by Testbed). 12 S: No brackets to then/else (added by Testbed). 26 S: Loop control expression may not terminate loop. 38 S: Use of static class member. 44 S: Use of banned function, type or variable. 49 S: Logical conjunctions need brackets. 57 S: Statement with no side effect. 62 S: Switch case not terminated with break. 99 S: Function use is not a call. 132 S: Assignment operator in boolean expression. 133 S: Assignment operator in RHS of && or . 136 S: Bit operator with boolean operand. 361 S: Expression needs brackets. 406 S: Use of ++ or -- on RHS of && or operator. 408 S: Volatile variable accessed on RHS of && or . 428 S: No {} for switch (added by Testbed). 527 S: No master exception handler. 583 S: Comparison of pointer and string literal. 592 S: Use of filename based functions. 1 U: Inter-file recursion found.		
693	Rule	Protection Mechanism Failure	This rule is a parent rule for 20, 37, 74, 77, 78, 88, 119, 120, 121, 122, 123, 124, 125, 126, 127, 129, 134, 138, 160, 170, 190, 269, 271, 273, 284, 327, 415, 416, 464, 466, 606, 786, 787, 788, 805, 806, 823, 824, 825, 1173 which has mappings to: 42 D: Local pointer returned in function result. 43 D: Divide by zero found. 51 D: Attempt to read from freed memory. 69 D: UR anomaly, variable used before assignment. 77 D: Local structure returned in function result. 80 D: Potentially unused function-modified value. 85 D: Filename not verified before fopen. 86 D: User input not checked before use. 91 D: Function return value potentially unused. 109 D: Tainted argument to formatted i/o function. 112 D: Free called twice on same variable. 124 D: Var set by std lib func return not checked before use. 127 D: Local or member denominator not checked before use. 130 D: Global set by std lib func return not checked before use. 131 D: Global denominator not checked within this procedure. 137 D: Parameter used as denominator not checked before use. 139 D: Incorrect array index validation. 140 D: Copy source parameter not checked before use. 141 D: Heap-based array index not checked before use. 44 S: Use of banned function, type or variable. 47 S: Array bound exceeded. 71 S: Pointer assignment to wider scope. 329 S: Operation not appropriate to plain char. 377 S: Null character used within a string. 382 S: (void) missing for discarded return value. 401 S: Use of sizeof on an array parameter. 403 S: Negative (or potentially negative) shift. 447 S: Loop termination not a simple constant test. 484 S: Attempt to use already freed object. 486 S: Incorrect number of formats in output function. 487 S: Insufficient space allocated. 488 S: Value outside range of underlying type. 489 S: Insufficient space for operation. 493 S: Numeric overflow. 503 S: Function returns local resources. 564 S: Reference assignment to wider scope. 567 S: Pointer arithmetic is not on array. 584 S: Remainder of % op could be negative. 588 S: Use of system function. 600 S: Argument of fopen is not terminated.		

			600 S: Argument or string is unterminated. 602 S: strtok may change the parse string. 629 S: Divide by zero found. 633 S: Use of a broken or risky cryptography algorithm. 687 S: Loop counter not checked for negative value. 688 S: Memory allocation exceeds 2 power *** bytes. 692 S: Array index is negative. 1 X: Declaration types do not match across a system. 64 X: Array bound exceeded at call. 66 X: Insufficient array space at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 70 X: Array has insufficient space. 71 X: Insufficient space for copy. 79 X: Size mismatch in memcpy/memset. 80 X: Divide by zero found.		
697	Rule	Incorrect Comparison	This rule is a parent rule for 187, 478, 595, 597, 1023, 1024, 1025, 1077 which has mappings to: 44 S: Use of banned function, type or variable. 48 S: No default case in switch statement. 56 S: Equality comparison of floating point. 101 S: Function return type inconsistent. 107 S: Type mismatch in ternary expression. 433 S: Type conversion without cast. 434 S: Signed/unsigned conversion without cast. 435 S: Float/integer conversion without cast. 458 S: Implicit conversion: actual to formal param (MR). 583 S: Comparison of pointer and string literal.	59 S	Else alternative missing in if. Modifiers :530 = 0 (Default)
703	Rule	Improper Check or Handling of Exceptional Conditions	This rule is a parent rule for 228, 229, 232, 233, 234, 235, 248, 252, 253, 273, 390, 391, 460, 476, 754, 755 which has mappings to: 45 D: Pointer not checked for null before use. 56 D: Throw found with no catch in scope. 71 D: No matching catch for throw in called function. 80 D: Potentially unused function-modified value. 91 D: Function return value potentially unused. 123 D: File pointer not checked for null before use. 124 D: Var set by std lib func return not checked before use. 128 D: Global pointer not checked within this procedure. 129 D: Global file pointer not checked within this procedure. 130 D: Global set by std lib func return not checked before use. 135 D: Pointer assigned to NULL may be dereferenced. 136 D: Global pointer assigned to NULL may be dereferenced. 21 S: Number of parameters does not match. 382 S: (void) missing for discarded return value. 527 S: No master exception handler. 531 S: Literal zero used in pointer context. 593 S: Use fseek() rather than rewind(). 594 S: Use setvbuf() rather than setbuf(). 634 S: Empty block found.		
704	Rule	Incorrect Type Conversion or Cast	This rule is a parent rule for 194, 195, 196, 197, 588, 681, 843 which has mappings to: 52 S: Unsigned expression negated. 101 S: Function return type inconsistent. 107 S: Type mismatch in ternary expression. 331 S: Literal value requires a U suffix. 403 S: Negative (or potentially negative) shift. 411 S: Inappropriate value assigned to enum. 433 S: Type conversion without cast. 434 S: Signed/unsigned conversion without cast. 435 S: Float/integer conversion without cast. 439 S: Cast from pointer to integral type. 440 S: Cast from integral type to pointer. 441 S: Float cast to non-float. 442 S: Signed integral type cast to unsigned. 443 S: Unsigned integral type cast to signed. 444 S: Integral type cast to non-integral. 446 S: Narrower int conversion without cast. 456 S: Implicit float widening for function return (MR). 458 S: Implicit conversion: actual to formal param (MR). 493 S: Numeric overflow. 540 S: Cast from pointer to void to pointer. 545 S: Assignment of overlapping storage.	203 S	Cast on a constant value. Modifiers :390 = 0 (Default)
				344 S	Cast on volatile value. Modifiers :395 = 0 (Default)
705	Rule	Incorrect Control Flow Scoping	This rule is a parent rule for 248 which has mappings to: 527 S: No master exception handler.		

706	Rule	Use of Incorrectly-Resolved Name or Reference	This rule is a parent rule for 22, 36, 37, 38, 39, 41, 59 which has mappings to: 85 D: Filename not verified before fopen.		
707	Rule	Improper Enforcement of Message or Data Structure	This rule is a parent rule for 37, 74, 77, 78, 88, 116, 138, 160, 170, 172, 176, 228, 229, 232, 233, 234, 235, 464 which has mappings to: 45 D: Pointer not checked for null before use. 85 D: Filename not verified before fopen. 21 S: Number of parameters does not match. 44 S: Use of banned function, type or variable. 377 S: Null character used within a string. 588 S: Use of system function. 600 S: Argument of strlen is unterminated. 602 S: strtok may change the parse string.		
			The LDRA toolsuite permits code to be checked against a wide range of coding standards. This rule is a parent rule for 14, 188, 242, 253, 415, 475, 476, 477, 484, 547, 560, 561, 562, 563, 570, 571, 573, 587, 588, 628, 675, 676, 685, 686, 687, 688, 733, 758, 1037, 1038, 1045, 1052, 1055, 1056, 1061, 1064, 1069, 1071, 1075, 1076, 1078, 1079, 1080, 1085, 1087, 1090, 1091, 1093, 1095, 1098, 1102, 1105, 1106, 1108, 1114, 1115, 1119, 1120, 1121, 1124, 1126, 1127, 1164, 1177 which has mappings to: 1 C: Cyclomatic complexity greater than ***. 10 C: Loop depth greater than ***. 2 D: Function does not return a value on all paths. 8 D: DD data flow anomalies found. 17 D: Identifier not unique within *** characters. 20 D: No declaration for variable found before use. 25 D: Scope of variable could be reduced. 35 D: Expression has side effects. 41 D: Procedure call has no prototype declared. 42 D: Local pointer returned in function result. 43 D: Divide by zero found. 45 D: Pointer not checked for null before use. 50 D: Memory not freed after last reference. 51 D: Attempt to read from freed memory. 53 D: Attempt to use uninitialised pointer. 55 D: Modification of loop counter in loop body. 63 D: No definition in system for prototyped procedure. 65 D: Void function has no side effects. 69 D: UR anomaly, variable used before assignment. 72 D: Potential side effect problem in expression. 75 D: Attempt to open file pointer more than once. 76 D: Procedure is not called or referenced in code analysed. 77 D: Local structure returned in function result. 78 D: Global variable should be declared const. 82 D: fseekpos values not generated by fseekpos. 83 D: Potentially repeated call to fseek. 84 D: No fseek or flush before I/O. 87 D: Illegal shared object in signal handler. 94 D: Named variable declared but not used in code. 105 D: DU anomaly dead code, var value is unused on all paths. 107 D: Attempt to change system call capture string. 112 D: Free called twice on same variable. 123 D: File pointer not checked for null before use. 125 D: free called on variable with no allocated space. 128 D: Global pointer not checked within this procedure. 129 D: Global file pointer not checked within this procedure. 135 D: Pointer assigned to NULL may be dereferenced. 136 D: Global pointer assigned to NULL may be dereferenced. 1 J: Unreachable Code found. 1 Q: Call has execution order dependant side effects. 5 Q: File does not end with new line. 3 S: More than *** executable reformatted lines in file. 5 S: Empty then clause. 7 S: Jump out of procedure. 13 S: goto detected. 18 S: More than *** parameters in procedure. 20 S: Parameter not declared explicitly. 21 S: Number of parameters does not match. 23 S: Procedure is not explicitly called in code analysed. 27 S: Void procedure with return statement. 35 S: Static procedure is not explicitly called in code analysed. 36 S: Function has no return statement. 41 S: Ellipsis used in procedure parameter list. 42 S: Use of bit field in structure declaration. 43 S: Use of setjmp/longjmp. 44 S: Use of banned function, type or variable. 47 S: Array bound exceeded. 51 S: Shifting value too far. 62 S: Switch case not terminated with break. 64 S: Void procedure used in expression. 65 S: Void variable passed as parameter. 66 S: Function with empty return expression. 71 S: Pointer assignment to wider scope. 73 S: Bit field not signed or unsigned int. 76 S: More than one of # or ## in a macro. 86 S: Attempt to define reserved word. 100 S: #include filename is non conformant.		

710	Rule	Improper Adherence to Coding Standards	105 S: Initialisation brace { } fault. 113 S: Non standard character in source. 115 S: String incorrectly terminated. 118 S: main must be int (void) or int (int,char*[]). 119 S: Nested comment found. 134 S: Volatile variable in complex expression. 135 S: Parameter list is KR. 139 S: Construct leads to infeasible code. 140 S: Infeasible loop condition found. 145 S: #if has invalid expression. 148 S: No return type for function/procedure. 157 S: Modification of string literal. 170 S: Procedure call has no prototype and no defn. 176 S: Non standard escape sequence in source. 184 S: Spaces round -> or [] operators. 185 S: Space between unary operator and operand. 190 S: { ... } contents not indented by *** spaces. 191 S: Space between function name and parenthesis. 201 S: Use of numeric literal in expression. 202 S: Class data is not explicitly private. 205 S: Use of multiple inheritance. 233 S: No copy constructor for class with pointers. 243 S: Included file not protected with #define. 248 S: Divide by zero in preprocessor directive. 262 S: Non virtual function redefined. 283 S: Multiple direct inheritance found. 296 S: Function declared at block scope. 302 S: Comment possibly contains code. 303 S: Virtual class members need virtual destructor. 315 S: Blank before/after . operator. 323 S: Switch has more than one default case. 324 S: Macro call has wrong number of parameters. 326 S: Declaration is missing type. 335 S: Operator defined contains illegal items. 336 S: #if expansion contains define operator. 339 S: #include directive with illegal items. 341 S: Preprocessor construct as macro parameter. 345 S: Bit operator with floating point operand. 355 S: Variables not unique within *** characters. 370 S: If nest depth greater than ***. 387 S: Enum init not integer-constant-expression. 392 S: Class data accessible thru non const member. 404 S: Array initialisation has too many items. 407 S: free used on string. 412 S: Undefined behaviour, \ before E-O-F. 427 S: Filename in #include not in < > or " ". 437 S: < > <= >= used on different object pointers. 438 S: Pointer subtraction not addressing one array. 440 S: Cast from integral type to pointer. 441 S: Float cast to non-float. 450 S: Wide string and string concatenated. 458 S: Implicit conversion: actual to formal param (MR). 465 S: Struct/union not completely specified. 480 S: String function params access same variable. 481 S: Array with no bounds in struct. 482 S: Incomplete structure referenced. 483 S: Freed parameter is not heap item. 484 S: Attempt to use already freed object. 486 S: Incorrect number of formats in output function. 487 S: Insufficient space allocated. 489 S: Insufficient space for operation. 493 S: Numeric overflow. 496 S: Function call with no prior declaration. 503 S: Function returns local resources. 520 S: Bit field is not bool or explicit integral. 531 S: Literal zero used in pointer context. 540 S: Cast from pointer to void to pointer. 545 S: Assignment of overlapping storage. 553 S: Function and proto should both be static. 554 S: Cast to an unrelated type. 560 S: Scope of variable could be reduced. 564 S: Reference assignment to wider scope. 565 S: Assignment to wider scope. 567 S: Pointer arithmetic is not on array. 568 S: #include "filename" uses standard library name. 573 S: Macro concatenation of uni char names. 575 S: Linkage differs from previous declaration. 576 S: Function pointer is of wrong type. 580 S: Macro redefinition without using #undef. 582 S: const object reassigned. 587 S: Const local variable not immediately initialised. 589 S: Format is not appropriate type. 590 S: Mode fault in fopen. 591 S: Inappropriate use of file pointer. 592 S: Use of filename based functions. 606 S: Cast involving function pointer. 608 S: Use of explicitly undefined language feature. 609 S: Use of implicitly undefined language feature. 615 S: Conditional operator has incompatible types. 617 S: Function call may return unexpected value. 618 S: Use of memcmp between structures. 623 S: String assigned to non const object. 629 S: Divide by zero found. 630 S: Duplicated enumeration value. 631 S: Declaration not reachable. 634 S: Empty block found. 635 S: Cast from pointer to float type. 636 S: Cast from float type to pointer. 642 S: Function return type with array field. 644 S: realloc ptr does not originate from allocation function. 645 S: realloc ptr type does not match target type. 646 S: Struct initialisation has too many items. 647 S: Overlapping data items in memcpy. 648 S: Deprecated form of flexible array. 649 S: Use of unallocated flexible array. 650 S: Flexible array member in struct.
-----	------	--	--

			650 S: Flexible array copy ignores last member. 652 S: Object created by malloc used before initialisation. 671 S: Class data accessible thru non const handle. 692 S: Array index is negative. 697 S: Pointer converted to different pointer type. 1 U: Inter-file recursion found. 1 X: Declaration types do not match across a system. 61 X: Identifier match in *** chars. 62 X: Function prototype/defn return type mismatch (MR). 63 X: Function prototype/defn param type mismatch (MR). 64 X: Array bound exceeded at call. 66 X: Insufficient array space at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 70 X: Array has insufficient space. 71 X: Insufficient space for copy. 72 X: Parameter indexing array too small at call. 79 X: Size mismatch in memcpy/memset. 80 X: Divide by zero found. 4 Z: Source does not match layout template.		
733	Rule	Compiler Optimization Removal or Modification of Security-critical Code	This rule is a parent rule for 14 which has mappings to: 105 D: DU anomaly dead code, var value is unused on all paths.		
754	Rule	Improper Check for Unusual or Exceptional Conditions	This rule is a parent rule for 252, 253, 273, 476 which has mappings to: 45 D: Pointer not checked for null before use. 80 D: Potentially unused function-modified value. 91 D: Function return value potentially unused. 123 D: File pointer not checked for null before use. 124 D: Var set by std lib func return not checked before use. 128 D: Global pointer not checked within this procedure. 129 D: Global file pointer not checked within this procedure. 130 D: Global set by std lib func return not checked before use. 135 D: Pointer assigned to NULL may be dereferenced. 136 D: Global pointer assigned to NULL may be dereferenced. 382 S: (void) missing for discarded return value. 531 S: Literal zero used in pointer context. 593 S: Use fseek() rather than rewind(). 594 S: Use setvbuf() rather than setbuf().		
755	Rule	Improper Handling of Exceptional Conditions	This rule is a parent rule for 390, 460 which has mappings to: 56 D: Throw found with no catch in scope. 71 D: No matching catch for throw in called function. 634 S: Empty block found.		
				2 D Function does not return a value on all paths. 17 D Identifier not unique within *** characters. 35 D Expression has side effects. 42 D Local pointer returned in function result. 43 D Divide by zero found. 45 D Pointer not checked for null before use. 51 D Attempt to read from freed memory. 53 D Attempt to use uninitialised pointer. 63 D No definition in system for prototyped procedure. 69 D UR anomaly, variable used before assignment. 72 D Potential side effect problem in expression. 77 D Local structure returned in function result. 82 D fsetpos values not generated by fgetpos.	

				83 D	Potentially repeated call to ungetc.
				84 D	No fseek or flush before I/O.
				87 D	Illegal shared object in signal handler.
				107 D	Attempt to change system call capture string.
				125 D	free called on variable with no allocated space.
				1 Q	Call has execution order dependant side effects.
				5 Q	File does not end with new line.
				36 S	Function has no return statement. Modifiers :553 = 0 (Default)
				47 S	Array bound exceeded.
				51 S	Shifting value too far. Modifiers :520 = 0 (Default)
				64 S	Void procedure used in expression. Modifiers :218 = 1
				65 S	Void variable passed as parameter.
				66 S	Function with empty return expression.
				71 S	Pointer assignment to wider scope. Modifiers :222 = 1
				73 S	Bit field not signed or unsigned int. Modifiers :234 = 0 (Default) 331 = 0 (Default) 350 = 1
				76 S	More than one of # or ## in a macro.
				86 S	Attempt to define reserved word.
				100 S	#include filename is non conformant. Modifiers :120 = \, (Default)
				105 S	Initialisation brace { } fault.
				113 S	Non standard character in source. Modifiers :408 = 1 449 = 0 (Default)
				115 S	String incorrectly terminated.
				118 S	main must be int (void) or int (int,char*[]).
				119 S	Nested comment found. Modifiers :413 = 0 (Default) 434 = 0 (Default)
				134 S	Volatile variable in complex expression. Modifiers :460 = 0 (Default) 461 = 1
				145 S	#if has invalid expression.
				157 S	Modification of string literal. Modifiers :438 = 1
				176 S	Non standard escape sequence in source.
				248 S	Divide by zero in preprocessor directive.
				296 S	Function declared at block scope.
				323 S	Switch has more than one default case.
				324 S	Macro call has wrong number of parameters.
				335 S	Operator defined contains illegal items.
				336 S	#if expansion contains define operator.
				339 S	#include directive with illegal items.
				341 S	Preprocessor construct as macro parameter.

758	Rule	Reliance on Undefined, Unspecified, or Implementation-Defined Behavior	This rule is a parent rule for 14, 188, 562, 587, 588, 733, 1037, 1038, 1102, 1105 which has mappings to: 76 D: Procedure is not called or referenced in code analysed. 105 D: DU anomaly dead code, var value is unused on all paths. 1 J: Unreachable Code found. 23 S: Procedure is not explicitly called in code analysed. 35 S: Static procedure is not explicitly called in code analysed. 42 S: Use of bit field in structure declaration. 139 S: Construct leads to infeasible code. 140 S: Infeasible loop condition found. 440 S: Cast from integral type to pointer. 503 S: Function returns local resources. 564 S: Reference assignment to wider scope. 567 S: Pointer arithmetic is not on array. 618 S: Use of memcmp between structures.	345 S	Bit operator with floating point operand.
				355 S	Variables not unique within *** characters.
				387 S	Enum init not integer-constant-expression.
				407 S	free used on string.
				412 S	Undefined behaviour, \ before E-O-F.
				427 S	Filename in #include not in < > or " ".
				437 S	< > <= >= used on different object pointers.
				438 S	Pointer subtraction not addressing one array.
				441 S	Float cast to non-float. Modifiers :502 = 1 503 = 0 (Default) 504 = 1 505 = 0 (Default) 506 = 1 529 = 1
				450 S	Wide string and string concatenated.
				465 S	Struct/union not completely specified.
				480 S	String function params access same variable.
				481 S	Array with no bounds in struct.
				482 S	Incomplete structure referenced.
				483 S	Freed parameter is not heap item.
				484 S	Attempt to use already freed object. Modifiers :350 = 1
				486 S	Incorrect number of formats in output function.
				487 S	Insufficient space allocated.
				489 S	Insufficient space for operation.
				493 S	Numeric overflow.
				520 S	Bit field is not bool or explicit integral. Modifiers :331 = 0 (Default) 350 = 1
				540 S	Cast from pointer to void to pointer. Modifiers :381 = 0 (Default)
				545 S	Assignment of overlapping storage.
				554 S	Cast to an unrelated type. Modifiers :439 = 0 (Default) 440 = 0 (Default)
				565 S	Assignment to wider scope.
				568 S	#include "filename" uses standard library name. Modifiers :344 = 1
				573 S	Macro concatenation of uni char names.
				575 S	Linkage differs from previous declaration.
				576 S	Function pointer is of wrong type.
				580 S	Macro redefinition without using #undef.
				582 S	const object reassigned.
				587 S	Const local variable not immediately initialised.
				589 S	Format is not appropriate type.
				590 S	Mode fault in fopen.

				591 S	Inappropriate use of file pointer.
				606 S	Cast involving function pointer. Modifiers :398 = 0 (Default) 433 = 0 (Default) 439 = 0 (Default)
				608 S	Use of explicitly undefined language feature.
				609 S	Use of implicitly undefined language feature.
				615 S	Conditional operator has incompatible types.
				617 S	Function call may return unexpected value.
				623 S	String assigned to non const object. Modifiers :119 = 0 (Default)
				629 S	Divide by zero found.
				635 S	Cast from pointer to float type.
				636 S	Cast from float type to pointer.
				642 S	Function return type with array field.
				644 S	realloc ptr does not originate from allocation function.
				645 S	realloc ptr type does not match target type.
				646 S	Struct initialisation has too many items.
				647 S	Overlapping data items in memcpy.
				648 S	Deprecated form of flexible array. Modifiers :464 = 1
				649 S	Use of unallocated flexible array.
				650 S	Flexible array copy ignores last member.
				652 S	Object created by malloc used before initialisation.
				692 S	Array index is negative.
				697 S	Pointer converted to different pointer type. Modifiers :441 = 0 (Default)
				1 X	Declaration types do not match across a system.
				61 X	Identifier match in *** chars.
				62 X	Function prototype/defn return type mismatch (MR).
				63 X	Function prototype/defn param type mismatch (MR).
				64 X	Array bound exceeded at call.
				66 X	Insufficient array space at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
				70 X	Array has insufficient space.
				71 X	Insufficient space for copy.
				72 X	Parameter indexing array too small at call.
				79 X	Size mismatch in memcpy/memset.
				80 X	Divide by zero found.

762	Rule	Mismatched Memory Management Routines	This rule is a parent rule for 590 which has mappings to: 125 D: free called on variable with no allocated space. 483 S: Freed parameter is not heap item. 644 S: realloc ptr does not originate from allocation function. The following standards are used to implement this rule	50 D	Memory not freed after last reference.
				407 S	free used on string.
				485 S	Array deletion without [].
763	Rule	Release of Invalid Pointer or Reference	This rule is a parent rule for 590, 762 which has mappings to: 50 D: Memory not freed after last reference. 125 D: free called on variable with no allocated space. 407 S: free used on string. 483 S: Freed parameter is not heap item. 485 S: Array deletion without []. 644 S: realloc ptr does not originate from allocation function.		
767	Rule	Access to Critical Private Variable via Public Method		392 S	Class data accessible thru non const member.
				671 S	Class data accessible thru non const handle.
768	Rule	Incorrect Short Circuit Evaluation		35 D	Expression has side effects.
				1 Q	Call has execution order dependant side effects.
				133 S	Assignment operator in RHS of && or .
				406 S	Use of ++ or -- on RHS of && or operator.
				408 S	Volatile variable accessed on RHS of && or .
772	Rule	Missing Release of Resource after Effective Lifetime	This rule is a parent rule for 401, 775, 1091 which has mappings to: 49 D: File pointer not closed on exit. 50 D: Memory not freed after last reference. 75 D: Attempt to open file pointer more than once. 81 D: Attempt to remove an open file.		
775	Rule	Missing Release of File Descriptor or Handle after Effective Lifetime		49 D	File pointer not closed on exit.
				75 D	Attempt to open file pointer more than once.
				81 D	Attempt to remove an open file.
783	Rule	Operator Precedence Logic Error	Standard 361 S is configured so that parentheses are not required for expressions containing only arithmetic operators.	49 S	Logical conjunctions need brackets. Modifiers :206 = 1
				361 S	Expression needs brackets. Modifiers :119 = 0 (Default) 264 = 1 414 = 0 (Default) 420 = 0 (Default) 421 = 1
786	Rule	Access of Memory Location Before Start of Buffer	This rule is a parent rule for 124, 127 which has mappings to: 47 S: Array bound exceeded. 584 S: Remainder of % op could be negative. 692 S: Array index is negative. 64 X: Array bound exceeded at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use.		
787	Rule	Out-of-bounds Write	This rule is a parent rule for 121, 122, 123, 124 which has mappings to: 86 D: User input not checked before use. 140 D: Copy source parameter not checked before use. 141 D: Heap-based array index not checked before use. 47 S: Array bound exceeded. 489 S: Insufficient space for operation. 584 S: Remainder of % op could be negative. 687 S: Loop counter not checked for negative value. 688 S: Memory allocation exceeds 2 power *** bytes. 692 S: Array index is negative. 64 X: Array bound exceeded at call. 66 X: Insufficient array space at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 70 X: Array has insufficient space. 71 X: Insufficient space for copy. 79 X: Size mismatch in memcpy/memset.		

788	Rule	Access of Memory Location After End of Buffer	This rule is a parent rule for 121, 122, 126 which has mappings to: 86 D: User input not checked before use. 140 D: Copy source parameter not checked before use. 141 D: Heap-based array index not checked before use. 47 S: Array bound exceeded. 489 S: Insufficient space for operation. 584 S: Remainder of % op could be negative. 687 S: Loop counter not checked for negative value. 688 S: Memory allocation exceeds 2 power *** bytes. 692 S: Array index is negative. 64 X: Array bound exceeded at call. 66 X: Insufficient array space at call. 68 X: Parameter indexing array too big at call. 69 X: Global array bound exceeded at use. 70 X: Array has insufficient space. 71 X: Insufficient space for copy. 79 X: Size mismatch in memcpy/memset.		
805	Rule	Buffer Access with Incorrect Length Value		109 D	Tainted argument to formatted i/o function.
				140 D	Copy source parameter not checked before use.
				489 S	Insufficient space for operation.
				66 X	Insufficient array space at call.
				70 X	Array has insufficient space.
				71 X	Insufficient space for copy.
				79 X	Size mismatch in memcpy/memset.
806	Rule	Buffer Access Using Size of Source Buffer		140 D	Copy source parameter not checked before use.
				489 S	Insufficient space for operation.
				66 X	Insufficient array space at call.
				70 X	Array has insufficient space.
				71 X	Insufficient space for copy.
				79 X	Size mismatch in memcpy/memset.
823	Rule	Use of Out-of-range Pointer Offset	567 S highlights all uses of pointer arithmetic. The user must then manually check whether the offset is out of range.	47 S	Array bound exceeded.
				567 S	Pointer arithmetic is not on array. Modifiers :436 = 0 (Default)
				584 S	Remainder of % op could be negative.
				692 S	Array index is negative.
				64 X	Array bound exceeded at call.
				68 X	Parameter indexing array too big at call.
				69 X	Global array bound exceeded at use.
824	Rule	Access of Uninitialized Pointer		69 D	UR anomaly, variable used before assignment.
825	Rule	Expired Pointer Dereference	This rule is a parent rule for 415, 416 which has mappings to: 51 D: Attempt to read from freed memory. 112 D: Free called twice on same variable. 484 S: Attempt to use already freed object.	42 D	Local pointer returned in function result.
				77 D	Local structure returned in function result.
				71 S	Pointer assignment to wider scope.
				503 S	Function returns local resources.
				564 S	Reference assignment to wider scope.
828	Rule	Signal Handler with Functionality that is not Asynchronous-Safe	This rule is a parent rule for 479 which has mappings to: 88 D: Illegal use of longjmp in signal handler. 89 D: Illegal use of raise in signal handler. 97 D: Signal called from within signal handler.		

834	Rule	Excessive Iteration	This rule is a parent rule for 835 which has mappings to: 5 C: Procedure contains infinite loop. 28 D: Potentially infinite loop found. 26 S: Loop control expression may not terminate loop.		
835	Rule	Loop with Unreachable Exit Condition ('Infinite Loop')		5 C	Procedure contains infinite loop.
				28 D	Potentially infinite loop found.
				26 S	Loop control expression may not terminate loop. Modifiers : 425 = 0 (Default)
843	Rule	Access of Resource Using Incompatible Type ('Type Confusion')	Other issues are likely to be picked up by 105 D (CWE-563) if one members of a union is assigned but not subsequently used.	545 S	Assignment of overlapping storage.
908	Rule	Use of Uninitialized Resource		69 D	UR anomaly, variable used before assignment.
909	Rule	Missing Initialization of Resource	This rule is a parent rule for 456 which has mappings to: 57 D: Global not initialised at declaration.		
910	Rule	Use of Expired File Descriptor		48 D	Attempt to write to unopened file.
				113 D	File closed more than once.
				126 D	File closed that was never opened.
1023	Rule	Incomplete Comparison with Missing Factors	This rule is a parent rule for 187, 478 which has mappings to: 44 S: Use of banned function, type or variable. 48 S: No default case in switch statement.		
1024	Rule	Comparison of Incompatible Types		107 S	Type mismatch in ternary expression. Modifiers :428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 470 = 0 (Default) 525 = 0 (Default)
				433 S	Type conversion without cast. Modifiers :374 = 0 (Default) 391 = 0 (Default) 431 = 1 428 = c : 0 (Default), c++ : 1 (Default) 470 = 0 (Default) 386 = 1 (Default) 549 = 0 (Default)
				434 S	Signed/unsigned conversion without cast. Modifiers :374 = 0 (Default) 358 = 1 427 = 0 (Default) 428 = c : 0 (Default), c++ : 1 (Default) 429 = 0 (Default) 430 = 0 (Default) 470 = 0 (Default)
				435 S	Float/integer conversion without cast. Modifiers :374 = 0 (Default) 432 = 0 (Default) 453 = 0 (Default)
				458 S	Implicit conversion: actual to formal param (MR).
1025	Rule	Comparison Using Wrong Factors		583 S	Comparison of pointer and string literal.
1037	Rule	Processor Optimization Removal or Modification of Security-critical Code	The user may have written code that is not reachable in normal situations, but is expected to catch issues such as hardware failures. The analysis highlights areas of code that are not reachable in normal situations and therefore might be candidates for optimisation by processors.	76 D	Procedure is not called or referenced in code analysed.
				1 J	Unreachable Code found.
				23 S	Procedure is not explicitly called in code analysed.
				35 S	Static procedure is not explicitly called in code analysed.
				139 S	Construct leads to infeasible code.
				140 S	Infeasible loop condition found.

1038	Rule	Insecure Automated Optimizations	This rule is a parent rule for 14, 733, 1037 which has mappings to: 76 D: Procedure is not called or referenced in code analysed. 105 D: DU anomaly dead code, var value is unused on all paths. 1 J: Unreachable Code found. 23 S: Procedure is not explicitly called in code analysed. 35 S: Static procedure is not explicitly called in code analysed. 139 S: Construct leads to infeasible code. 140 S: Infeasible loop condition found.		
1042	Rule	Static Member Data Element outside of a Singleton Class Element	The analysis highlights all uses of static class member. The user must then check whether the requirements of this rule are met.	38 S	Use of static class member. Modifiers :423 = 1
1045	Rule	Parent Class with a Virtual Destructor and a Child Class without a Virtual Destructor		262 S	Non virtual function redefined. Modifiers :379 = 0 (Default)
				303 S	Virtual class members need virtual destructor. Modifiers :292 = 1
1048	Rule	Invokable Control Element with Large Number of Outward Calls	Details of the fan-out of each procedure is recorded in the Quality Review Report.		
1052	Rule	Excessive Use of Hard-Coded Literals in Initialization	The analysis highlights the use of all numeric literals values (except 0, 1 and those literals used in #define and as enumeration constants).	201 S	Use of numeric literal in expression. Modifiers :217 = 1 163 = 1 (Default) 237 = 0 (Default) 207 = 0 (Default)
1055	Rule	Multiple Inheritance from Concrete Classes	The analysis highlights all uses of multiple inheritance, but standards 205 S is not limited to concrete (non-abstract) classes. 238 S is configured with modifier 378 to permit various forms of direct inheritance	205 S	Use of multiple inheritance.
				283 S	Multiple direct inheritance found. Modifiers :378 = 1
1056	Rule	Invokable Control Element with Variadic Parameters		41 S	Ellipsis used in procedure parameter list. Modifiers :522 = 1
1058	Rule	Invokable Control Element in Multi-Thread Context with non-Final Static Storable or Member Element	The analysis highlights all uses of static class member. The user must then check whether the requirements of this rule are met.	38 S	Use of static class member.
1061	Rule	Insufficient Encapsulation	This rule is a parent rule for 188, 1090, 1105 which has mappings to: 392 S: Class data accessible thru non const member. 567 S: Pointer arithmetic is not on array. 618 S: Use of memcmp between structures. 671 S: Class data accessible thru non const handle.	202 S	Class data is not explicitly private.
1064	Rule	Invokable Control Element with Signature Containing an Excessive Number of Parameters	The analysis highlights functions with more than 7 parameters, but this can be altered by a project.	18 S	More than *** parameters in procedure. Modifiers : 209 = 0 (Default)
1069	Rule	Empty Exception Block	The analysis does not permits comments to count as a non-empty block. However comments may be permitted by setting the modifier 422 to 1.	634 S	Empty block found. Modifiers :422 = 1
1071	Rule	Empty Code Block	The analysis does not permits comments to count as a non-empty block. However comments may be permitted by setting the modifier 422 to 1.	5 S	Empty then clause.
				634 S	Empty block found. Modifiers :422 = 1
1074	Rule	Class with Excessively Deep Inheritance	Details of the depth of the inheritance is recorded in the Quality Review Report.		
1075	Rule	Unconditional Control Flow Transfer outside of Switch Block		7 S	Jump out of procedure.
				13 S	goto detected.

1076	Rule	Insufficient Adherence to Expected Conventions	This rule is a parent rule for 547, 1045, 1052, 1078, 1079, 1085, 1087, 1091, 1098, 1106, 1108, 1114, 1115 which has mappings to: 50 D: Memory not freed after last reference. 78 D: Global variable should be declared const. 184 S: Spaces round -> or [] operators. 185 S: Space between unary operator and operand. 190 S: { ... } contents not indented by *** spaces. 191 S: Space between function name and parenthesis. 201 S: Use of numeric literal in expression. 233 S: No copy constructor for class with pointers. 262 S: Non virtual function redefined. 302 S: Comment possibly contains code. 303 S: Virtual class members need virtual destructor. 315 S: Blank before/after . operator. 4 Z: Source does not match layout template.		
1077	Rule	Floating Point Comparison with Incorrect Operator		56 S	Equality comparison of floating point. Modifiers :223 = 1
1078	Rule	Inappropriate Source Code Style or Formatting	This rule is a parent rule for 547, 1052, 1085, 1106, 1114, 1115 which has mappings to: 184 S: Spaces round -> or [] operators. 185 S: Space between unary operator and operand. 190 S: { ... } contents not indented by *** spaces. 191 S: Space between function name and parenthesis. 201 S: Use of numeric literal in expression. 302 S: Comment possibly contains code. 315 S: Blank before/after . operator. 4 Z: Source does not match layout template.		
1079	Rule	Parent Class without Virtual Destructor Method		262 S	Non virtual function redefined. Modifiers :379 = 0 (Default)
				303 S	Virtual class members need virtual destructor. Modifiers :292 = 1
1080	Rule	Source Code File with Excessive Number of Lines of Code	The analysis counts the number of reformatted lines of code. The value is set to 1000, but can be configured by the user.	3 S	More than *** executable reformatted lines in file.
1085	Rule	Invokable Control Element with Excessive Volume of Commented-out Code	The analysis highlights all uses of commented out code. The user must then manually check whether it meets the requirements of the project.	302 S	Comment possibly contains code.
1086	Rule	Class with Excessive Number of Child Classes	Details of the number of base classes and depth of the inheritance is recorded in the Quality Review Report.		
1087	Rule	Class with Virtual Method without a Virtual Destructor		303 S	Virtual class members need virtual destructor. Modifiers :292 = 1
1090	Rule	Method Containing Access of a Member Element from Another Class		392 S	Class data accessible thru non const member.
				671 S	Class data accessible thru non const handle.
1091	Rule	Use of Object without Invoking Destructor Method	The analysis highlights those cases where the object has been created using new, but there has been no call of delete	50 D	Memory not freed after last reference.
1093	Rule	Excessively Complex Data Representation	This rule is a parent rule for 1055 which has mappings to: 205 S: Use of multiple inheritance. 283 S: Multiple direct inheritance found.		
1095	Rule	Loop Condition Value Update within the Loop		55 D	Modification of loop counter in loop body.
1098	Rule	Data Element containing Pointer Item without Proper Copy Control Element		233 S	No copy constructor for class with pointers.
1099	Rule	Inconsistent Naming Conventions for Identifiers	The analysis can be configured to check for various naming conventions of variables. Teese can be found in the style section of the user documentation. Instructions for enabling the H standards can be found in H_standards folder of the Testbed installation directory. The code will need to be altered by the user so that the required prefixes can be checked.		
1102	Rule	Reliance on Machine-Dependent Data Representation		42 S	Use of bit field in structure declaration.
1105	Rule	Insufficient Encapsulation of Machine-Dependent Functionality	This rule is a parent rule for 188 which has mappings to: 567 S: Pointer arithmetic is not on array. 618 S: Use of memcmp between structures.		

1106	Rule	Insufficient Use of Symbolic Constants	The analysis highlights the use of all numeric literals values (except 0, 1 and those literals used in #define and as enumeration constants).	201 S	Use of numeric literal in expression. Modifiers :217 = 1 163 = 1 (Default) 237 = 0 (Default) 207 = 0 (Default)
1108	Rule	Excessive Reliance on Global Variables	The analysis highlights all uses of global variables which are not declared as constants.	78 D	Global variable should be declared const.
1114	Rule	Inappropriate Whitespace Style	The analysis can be configured to meet the whitespace style chosen by the project.	184 S	Spaces round -> or [] operators. Modifiers :240 = 0 (Default)
				185 S	Space between unary operator and operand.
				190 S	{ ... } contents not indented by *** spaces. Modifiers :324 = 0 (Default) 518 = 0 (Default)
				191 S	Space between function name and parenthesis.
				315 S	Blank before/after . operator.
1115	Rule	Source Code Element without Standard Prologue	The project can be configured to check a template at the start of each file. Instructions for enabling the Z standards can be found in User_standards folder of the Testbed installation directory.	4 Z	Source does not match layout template.
1119	Rule	Excessive Use of Unconditional Branching	The analysis highlights all uses of goto and jmp routines. The user must then check whether the number meets the projects requirements.	13 S	goto detected.
				43 S	Use of setjmp/longjmp.
1120	Rule	Excessive Code Complexity	This rule is a parent rule for 1056, 1064, 1075, 1080, 1095, 1119, 1121, 1124 which has mappings to: 1 C: Cyclomatic complexity greater than ***. 10 C: Loop depth greater than ***. 55 D: Modification of loop counter in loop body. 3 S: More than *** executable reformatted lines in file. 7 S: Jump out of procedure. 13 S: goto detected. 18 S: More than *** parameters in procedure. 41 S: Ellipsis used in procedure parameter list. 43 S: Use of setjmp/longjmp. 370 S: If nest depth greater than ***.		
1121	Rule	Excessive McCabe Cyclomatic Complexity	By default the complexity is set to 10. Instructions for changing the value can be found in the c/cppreport.dat file.	1 C	Cyclomatic complexity greater than ***.
1122	Rule	Excessive Halstead Complexity	Details of halstead Complexity can be found in the Quality Review Report.		
1124	Rule	Excessively Deep Nesting	By default the if depth is set to 4, loop depth to 3. Instructions for changing the value can be found in the c/cppreport.dat file	10 C	Loop depth greater than ***.
				370 S	If nest depth greater than ***.
1126	Rule	Declaration of Variable with Unnecessarily Wide Scope		25 D	Scope of variable could be reduced.
				560 S	Scope of variable could be reduced.
				20 D	No declaration for variable found before use.
				21 S	Number of parameters does not match.
				27 S	Void procedure with return statement.
				145 S	#if has invalid expression.
				243 S	Included file not protected with #define.
				323 S	Switch has more than one default case.
				345 S	Bit operator with floating point operand.

1127	Rule	Compilation with Insufficient Warnings or Errors	The analysis should be performed after compilation by the user's compiler. The standards mapped to this rule protect from the following constraint errors, which have been known to go undetected by some compilers. In addition, some common warnings of poor code are also mapped to this rule.	387 S	Enum init not integer-constant-expression.
				404 S	Array initialisation has too many items. Modifiers :399 = 1
				481 S	Array with no bounds in struct.
				553 S	Function and proto should both be static.
				580 S	Macro redefinition without using #undef.
				615 S	Conditional operator has incompatible types.
				630 S	Duplicated enumeration value.
				645 S	realloc ptr type does not match target type.
				646 S	Struct initialisation has too many items.
1164	Rule	Irrelevant Code	This rule is a parent rule for 561, 563, 1069, 1071 which has mappings to: 8 D: DD data flow anomalies found. 65 D: Void function has no side effects. 76 D: Procedure is not called or referenced in code analysed. 94 D: Named variable declared but not used in code. 105 D: DU anomaly dead code, var value is unused on all paths. 1 J: Unreachable Code found. 5 S: Empty then clause. 23 S: Procedure is not explicitly called in code analysed. 35 S: Static procedure is not explicitly called in code analysed. 139 S: Construct leads to infeasible code. 140 S: Infeasible loop condition found. 631 S: Declaration not reachable. 634 S: Empty block found. 1 U: Inter-file recursion found.		
1173	Rule	Improper Use of Validation Framework		86 D	User input not checked before use.
1176	Rule	Inefficient CPU Computation	This rule is a parent rule for 1042 which has mappings to: 38 S: Use of static class member.		
1177	Rule	Use of Prohibited Code	The analysis can be configured to highlight functions prohibited by the project. 44 S gives instructions for adding project defined functions. This rule is a parent rule for 242, 676 which has mappings to: 592 S: Use of filename based functions.	44 S	Use of banned function, type or variable.
1187	Rule	Use of Uninitialized Resource	This rule is a parent rule for 457 which has mappings to: 53 D: Attempt to use uninitialised pointer. 54 D: Unsafe use of function pointer variable. 631 S: Declaration not reachable. 652 S: Object created by malloc used before initialisation.	69 D	UR anomaly, variable used before assignment.

General Compliance Notes

Enhanced Enforcement: LDRA checks additional cases to those specified by the mapped rule for enhanced safety and security.

Fully Implemented: LDRA checks all statically checkable aspects of the mapped rule.

Partially Implemented: LDRA checks certain aspects of the rule.

The assessment of whether a rule is fully or partially implemented is based on whether the mapped LDRA standards cover all statically checkable aspects of the rule with a high level of coverage or only cover certain statically checkable aspects of the rule. If a rule is undecidable then this assessment is based on what it is deemed reasonable for a static analysis tool to check.