

# ISO/IEC TS 17961:2013 - C Secure

## Perforce QAC for C 2026.1

(ISO/IEC TS 17961:2013 - C Secure ISO/IEC TS 17961 Information Technology - Programming languages, their environments and system software interfaces - C Secure Coding Rules)

### Rule Enforcement Summary

		Total
a	Total Number of Rules	46
b	Total Number of 'Not Statically Enforceable' Rules (Assisted/Unassisted)	0
c	Total Number of Enforceable Rules (a-b)	46
d	Total Number of Enforced Rules	45
e	Total Number of Unenforced Rules (c-d)	1
f	Enforced Rules Percentage (d/c)	98%
g	Unenforced Rules Percentage (e/c)	2%

Id	Description	Enforced
Rule-5.1	Accessing an object through a pointer to an incompatible type [ptrcomp]	Yes
Rule-5.2	Accessing freed memory [accfree]	Yes
Rule-5.3	Accessing shared objects in signal handlers [accsig]	Yes
Rule-5.4	No assignment in conditional expressions [boolasgn]	Yes
Rule-5.5	Calling functions in the C Standard Library other than abort, _Exit, and signal from within a signal handler [asynsig]	Yes
Rule-5.6	Calling functions with incorrect arguments [argcomp]	Yes
Rule-5.7	Calling signal from interruptible signal handlers [sigcall]	Yes
Rule-5.8	Calling system [syscall]	Yes
Rule-5.9	Comparison of padding data [padcomp]	Yes

Id	Description	Enforced
Rule-5.10	Converting a pointer to integer or integer to pointer [intptrconv]	Yes
Rule-5.11	Converting pointer values to more strictly aligned pointer types [alignconv]	Yes
Rule-5.12	Copying a FILE object [filecpy]	Yes
Rule-5.13	Declaring the same function or object in incompatible ways [funcdecl]	Yes
Rule-5.14	Dereferencing an out-of-domain pointer [nullref]	Yes
Rule-5.15	Escaping of the address of an automatic object [addrescape]	Yes
Rule-5.16	Conversion of signed characters to wider integer types before a check for EOF [signconv]	Yes
Rule-5.17	Use of an implied default in a switch statement [swtchdflt]	Yes
Rule-5.18	Failing to close files or free dynamic memory when they are no longer needed [fileclose]	Yes
Rule-5.19	Failing to detect and handle standard library errors [liberr]	Yes
Rule-5.20	Forming invalid pointers by library function [libptr]	Yes
Rule-5.21	Allocating insufficient memory [insufmem]	Yes
Rule-5.22	Forming or using out-of-bounds pointers or array subscripts [invptr]	Yes
Rule-5.23	Freeing memory multiple times [dblfree]	Yes
Rule-5.24	Including tainted or out-of-domain input in a format string [usrfmt]	Yes
Rule-5.25	Incorrectly setting and using errno [inverrno]	Yes
Rule-5.26	Integer division errors [diverr]	Yes
Rule-5.27	Interleaving stream inputs and outputs without a flush or positioning call [ioileave]	Yes
Rule-5.28	Modifying string literals [strmod]	Yes
Rule-5.29	Modifying the string returned by getenv, localeconv, setlocale, and strerror [libmod]	Yes
Rule-5.30	Overflowing signed integers [intoflow]	Yes
Rule-5.31	Passing a non-null-terminated character sequence to a library function that expects a string [nonnullcs]	Yes

Id	Description	Enforced
Rule-5.32	Passing arguments to character-handling functions that are not representable as unsigned char [chrsgnext]	Yes
Rule-5.33	Passing pointers into the same object as arguments to different restrict-qualified parameters [restrict]	Yes
Rule-5.34	Reallocating or freeing memory that was not dynamically allocated [xfree]	Yes
Rule-5.35	Referencing uninitialized memory [uninitref]	Yes
Rule-5.36	Subtracting or comparing two pointers that do not refer to the same array [ptrobj]	Yes
Rule-5.37	Tainted strings are passed to a string copying function [taintstrcpy]	Yes
Rule-5.38	Taking the size of a pointer to determine the size of the pointed-to type [sizeofptr]	Yes
Rule-5.39	Using a tainted value as an argument to an unprototyped function pointer [taintnoproto]	No
Rule-5.40	Using a tainted value to write to an object using a formatted input or output function [taintformatio]	Yes
Rule-5.41	Using a value for fsetpos other than a value returned from fgetpos [xfilepos]	Yes
Rule-5.42	Using an object overwritten by getenv, localeconv, setlocale, and strerror [libuse]	Yes
Rule-5.43	Using character values that are indistinguishable from EOF [chreof]	Yes
Rule-5.44	Using identifiers that are reserved for the implementation [resident]	Yes
Rule-5.45	Using invalid format strings [invfmtstr]	Yes
Rule-5.46	Tainted, potentially mutilated, or out-of-domain integer values are used in a restricted sink [taintsink]	Yes

Rule text copyright ISO 2025.