Security Weakness Descriptions

This document presents descriptions of the 22 weaknesses contained in the CISQ Automated Quality Characteristic Measure for Security. These descriptions have been simplified from their description in the published OMG® specification that used formalisms from other OMG meta-models to specify the weaknesses for representation in machine-processable XMI notation. The tables below present each weakness with its unique CISQ identifier, a brief descriptive name, and a fuller description of the weakness presented as a recommendation for remediation.

Security Weaknesses

a measure of the extent to which software contains weaknesses that can be exploited to gain unauthorized access to a system to steal data, cause damage, or other malicious acts.

CISQ identifier	Descriptor	Remediation
ASCSM-CWE-22	Improper path traversal	Remove instances where a user input is ultimately used in a file path creation statement, without any sanitization (based on a list of vetted sanitization functions, methods, procedures, stored procedures, sub-routines, etc.) of the user input value between the user input and the statement.
ASCSM-CWE-78	OS command injection	Remove instances where a user input is ultimately used to execute an OS command, without any sanitization (based on a list of vetted sanitization functions, methods, procedures, stored procedures, sub-routines, etc.) of the user input value between the user input and the statement.
ASCSM-CWE-79	Cross-site scripting	Remove instances where a user input is ultimately displayed to the user, without any sanitization (based on a list of vetted sanitization functions, methods, procedures, stored procedures, sub-routines, etc.) of the user input value between the user input and the statement.
ASCSM-CWE-89	SQL injection	Remove instances where a user input is ultimately used in a SQL statement, without any sanitization (based on a list of vetted sanitization functions, methods, procedures, stored procedures, sub-routines, etc.) of the user input value between the user input and the statement.

ASCSM-CWE-99	Unsanitized user input used to access a named resource	Remove instances where a user input is ultimately used to access a resource by name, without any sanitization (based on a list of vetted sanitization functions, methods, procedures, stored procedures, sub-routines, etc.) of the user input value between the user input and the statement.
ASCSM-CWE-120	Buffer overflow	Remove instances where the content of the first buffer is moved into the content of the second buffer while their allocated sizes are incompatible
ASCSM-CWE-129	Unchecked array index range	Remove instances where a user input is ultimately used in a 'Read' or 'Write' access to an array, without any range check between the user input and the array access.
ASCSM-CWE-134	Improper format string neutralization	Remove instances where a user input is ultimately used in a formatting statement, without any sanitization (based on a list of vetted sanitization functions, methods, procedures, stored procedures, sub-routines, etc.) of the user input value between the user input and the statement.
ASCSM-CWE-252-resource	Unchecked return parameter from resource handling operations	Remove instances where the function, method, procedure, stored procedure, sub-routine, etc. reads, writes, or manages an external resource, yet the value of the return code is not checked anywhere
ASCSM-CWE-327	Unvetted cryptographic algorithms	Remove instances where the application uses a cryptographic list which is not part of the list of vetted cryptographic libraries.
ASCSM-CWE-396	Catch of overly broad exception types	Remove instances where the function, method, procedure, stored procedure, sub-routine, etc. contains a catch which declares to catch an exception whose type is part of a list of overly broad exception types
ASCSM-CWE-397	Throw of overly broad exception types	Remove instances where the function, method, procedure, stored procedure, sub-routine, etc. throws an exception whose type is part of a list of overly broad exception types
ASCSM-CWE-434	Unsanitized user input in file upload statement	Remove instances where a user input is ultimately used in a file upload statement, without any sanitization (based on a list of vetted sanitization functions, methods, procedures, stored procedures, sub-routines, etc.) of the user input value between the user input and the statement.
ASCSM-CWE-456	Uninitialized data element	Remove instances where a variable, field, member, etc. is declared, then is evaluated without ever being initialized prior to the evaluation.
ASCSM-CWE-606	Unchecked input in loop condition	Remove instances where a user input is ultimately used in the loop condition statement, without any range check between the user input and the loop statement.

ASCSM-CWE-667	Improper locking of shared resources	Remove instances where the shared variable, field, member, etc., is accessed outside a critical section of the application.
ASCSM-CWE-672	Access to released or expired resources	Remove instances where the platform resource (messaging, lock, file, stream, etc.) is deallocated using its unique resource handler which is used later within the application to try and access the resource.
ASCSM-CWE-681	Incompatible numeric type conversion	Remove instances where a variable, field, member, etc. is declared with a numeric type, and then is updated with a value from a second numeric type that is incompatible with the first numeric type
ASCSM-CWE-772	Unreleased resource	Remove instances where a platform resource (CPU, messaging, lock, file, stream, etc.) is allocated and assigned a unique resource handler, and its unique resource handler is used throughout the application along a sequence of operations, but none of which is a release statement
ASCSM-CWE-789	Unchecked range of user input to a buffer	Remove instances where a user input is ultimately used in a 'Read' or 'Write' access to a buffer, without any range check between the user input and the buffer access.
ASCSM-CWE-798	Hard-coded credentials for remote resources	Remove instances where a variable, field, member, etc., is initialized with a hard-coded literal value, and ultimately used to access a remote resource.
ASCSM-CWE-835	Infinite recursion	Remove instances where a recursive function, method, procedure, stored procedure, sub-routine, etc., has no execution path to exit the recursion