Integer Overflow

Return to Glossary

An **Integer Overflow** is the result of trying to place into computer memory an integer (whole number) that is too large for the integer data type in a given system. For example, if an integer data type allows integers up to two bytes or 16 bits in length (or an unsigned number up to decimal 65,535), and two integers are to be added together that will exceed the value of 65,535, the result will be integer overflow. According to ISO C99, the C programming language standard, the actual value resulting from an instance of integer overflow must be regarded as unpredictable. (In practice, integer overflow usually results in a “wrap-around” value where the addition of 1 to the maximum value results in a value of 0.)

Integer overflow can result, for example, in a request for dynamically allocated memory that is far too large or too small needed by the program.

> **Note:** An integer overflow often passes undetected by the affected application. Because of this, the condition may lead to a security breach through a buffer overflow or other malicious code.

Source: https://searchsoftwarequality.techtarget.com/definition/integer-overflow

From:
https://omgwiki.org/dido/ - **DIDO Wiki**

Permanent link:
https://omgwiki.org/dido/doku.php?id=dido:public:ra:xapend:xapend.a_glossary:o:overflow&rev=1640643339

Last update: **2021/12/27 17:15**