# CWE Detail – CWE-193

## Description

A product calculates or uses an incorrect maximum or minimum value that is 1 more, or 1 less, than the correct value.

## Extended Description

N/A

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2003-0252: Off-by-one error allows remote attackers to cause a denial of service and possibly execute arbitrary code via requests that do not contain newlines.

**•** CVE-2001-1391: Off-by-one vulnerability in driver allows users to modify kernel memory.

**•** CVE-2002-0083: Off-by-one error allows local users or remote malicious servers to gain privileges.

**•** CVE-2002-0653: Off-by-one buffer overflow in function usd by server allows local users to execute arbitrary code as the server user via .htaccess files with long entries.

**•** CVE-2002-0844: Off-by-one buffer overflow in version control system allows local users to execute arbitrary code.

**•** CVE-1999-1568: Off-by-one error in FTP server allows a remote attacker to cause a denial of service (crash) via a long PORT command.

**•** CVE-2004-0346: Off-by-one buffer overflow in FTP server allows local users to gain privileges via a 1024 byte RETR command.

**•** CVE-2004-0005: Multiple buffer overflows in chat client allow remote attackers to cause a denial of service and possibly execute arbitrary code.

**•** CVE-2003-0356: Multiple off-by-one vulnerabilities in product allow remote attackers to cause a denial of service and possibly execute arbitrary code.

**•** CVE-2001-1496: Off-by-one buffer overflow in server allows remote attackers to cause a denial of service and possibly execute arbitrary code.

**•** CVE-2004-0342: This is an interesting example that might not be an off-by-one.

**•** CVE-2001-0609: An off-by-one enables a terminating null to be overwritten, which causes 2 strings to be merged and enable a format string.

**•** CVE-2002-1745: Off-by-one error allows source code disclosure of files with 4 letter extensions that match an accepted 3-letter extension.

**•** CVE-2002-1816: Off-by-one buffer overflow.

**•** CVE-2002-1721: Off-by-one error causes an snprintf call to overwrite a critical internal variable with a null value.

**•** CVE-2003-0466: Off-by-one error in function used in many products leads to a buffer overflow during pathname management, as demonstrated using multiple commands in an FTP server.

**•** CVE-2003-0625: Off-by-one error allows read of sensitive memory via a malformed request.

**•** CVE-2006-4574: Chain: security monitoring product has an off-by-one error that leads to unexpected length values, triggering an assertion.

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: DoS: Crash, Exit, or Restart, DoS: Resource Consumption (CPU), DoS: Resource Consumption (Memory), DoS: Instability — Notes: This weakness will generally lead to undefined behavior and therefore crashes. In the case of overflows involving loop index variables, the likelihood of infinite loops is also high.

**•** Impact: Modify Memory — Notes: If the value in question is important to data (as opposed to flow), simple data corruption has occurred. Also, if the wrap around results in other conditions such as buffer overflows, further memory corruption may occur.

**•** Impact: Execute Unauthorized Code or Commands, Bypass Protection Mechanism — Notes: This weakness can sometimes trigger buffer overflows which can be used to execute arbitrary code. This is usually outside the scope of a program's implicit security policy.

## Potential Mitigations

**•** Implementation: When copying character arrays or using character manipulation methods, the correct size parameter must be used to account for the null terminator that needs to be added at the end of the array. Some examples of functions susceptible to this weakness in C include strcpy(), strncpy(), strcat(), strncat(), printf(), sprintf(), scanf() and sscanf(). (Effectiveness: N/A)

## Applicable Platforms

**•** C (Class: None, Prevalence: Undetermined)

**•** None (Class: Not Language-Specific, Prevalence: Undetermined)

## Demonstrative Examples

**•** However, this code contains an off-by-one calculation error (CWE-193). It allocates exactly enough space to contain the specified number of widgets, but it does not include the space for the NULL pointer. As a result, the allocated buffer is smaller than it is supposed to be (CWE-131). So if the user ever requests MAX\_NUM\_WIDGETS, there is an out-of-bounds write (CWE-787) when the NULL is assigned. Depending on the environment and compilation settings, this could cause memory corruption.

**•** The first call to strncat() appends up to 20 characters plus a terminating null character to fullname[]. There is plenty of allocated space for this, and there is no weakness associated with this first call. However, the second call to strncat() potentially appends another 20 characters. The code does not account for the terminating null character that is automatically added by strncat(). This terminating null character would be written one byte beyond the end of the fullname[] buffer. Therefore an off-by-one error exists with the second strncat() call, as the third argument should be 19.

**•** If i reaches PATH\_SIZE, then the loop continues. However, filename[PATH\_SIZE] is actually out of bounds, since the valid index range is from 0 to PATH\_SIZE-1.

**•** However this will cause an Off-by-one error if the original filename is exactly 16 characters or larger because the format of 16 characters with the file extension is exactly 20 characters and does not take into account the required null terminator that will be placed at the end of the string.

## Notes

**•** Relationship: This is not always a buffer overflow. For example, an off-by-one error could be a factor in a partial comparison, a read from the wrong memory location, an incorrect conditional, etc.