# CWE Detail – CWE-476

## Description

The product dereferences a pointer that it expects to be valid but is NULL.

## Extended Description

N/A

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2005-3274: race condition causes a table to be corrupted if a timer activates while it is being modified, leading to resultant NULL dereference; also involves locking.

**•** CVE-2002-1912: large number of packets leads to NULL dereference

**•** CVE-2005-0772: packet with invalid error status value triggers NULL dereference

**•** CVE-2009-4895: Chain: race condition for an argument value, possibly resulting in NULL dereference

**•** CVE-2020-29652: ssh component for Go allows clients to cause a denial of service (nil pointer dereference) against SSH servers.

**•** CVE-2009-2692: Chain: Use of an unimplemented network socket operation pointing to an uninitialized handler function (CWE-456) causes a crash because of a null pointer dereference (CWE-476).

**•** CVE-2009-3547: Chain: race condition (CWE-362) might allow resource to be released before operating on it, leading to NULL dereference (CWE-476)

**•** CVE-2009-3620: Chain: some unprivileged ioctls do not verify that a structure has been initialized before invocation, leading to NULL dereference

**•** CVE-2009-2698: Chain: IP and UDP layers each track the same value with different mechanisms that can get out of sync, possibly resulting in a NULL dereference

**•** CVE-2009-2692: Chain: uninitialized function pointers can be dereferenced allowing code execution

**•** CVE-2009-0949: Chain: improper initialization of memory can lead to NULL dereference

**•** CVE-2008-3597: Chain: game server can access player data structures before initialization has happened leading to NULL dereference

**•** CVE-2020-6078: Chain: The return value of a function returning a pointer is not checked for success (CWE-252) resulting in the later use of an uninitialized variable (CWE-456) and a null pointer dereference (CWE-476)

**•** CVE-2008-0062: Chain: a message having an unknown message type may cause a reference to uninitialized memory resulting in a null pointer dereference (CWE-476) or dangling pointer (CWE-825), possibly crashing the system or causing heap corruption.

**•** CVE-2008-5183: Chain: unchecked return value can lead to NULL dereference

**•** CVE-2004-0079: SSL software allows remote attackers to cause a denial of service (crash) via a crafted SSL/TLS handshake that triggers a null dereference.

**•** CVE-2004-0365: Network monitor allows remote attackers to cause a denial of service (crash) via a malformed RADIUS packet that triggers a null dereference.

**•** CVE-2003-1013: Network monitor allows remote attackers to cause a denial of service (crash) via a malformed Q.931, which triggers a null dereference.

**•** CVE-2003-1000: Chat client allows remote attackers to cause a denial of service (crash) via a passive DCC request with an invalid ID number, which causes a null dereference.

**•** CVE-2004-0389: Server allows remote attackers to cause a denial of service (crash) via malformed requests that trigger a null dereference.

**•** CVE-2004-0119: OS allows remote attackers to cause a denial of service (crash from null dereference) or execute arbitrary code via a crafted request during authentication protocol selection.

**•** CVE-2004-0458: Game allows remote attackers to cause a denial of service (server crash) via a missing argument, which triggers a null pointer dereference.

**•** CVE-2002-0401: Network monitor allows remote attackers to cause a denial of service (crash) or execute arbitrary code via malformed packets that cause a NULL pointer dereference.

**•** CVE-2001-1559: Chain: System call returns wrong value (CWE-393), leading to a resultant NULL dereference (CWE-476).

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: DoS: Crash, Exit, or Restart — Notes: NULL pointer dereferences usually result in the failure of the process unless exception handling (on some platforms) is available and implemented. Even when exception handling is being used, it can still be very difficult to return the software to a safe state of operation.

**•** Impact: Execute Unauthorized Code or Commands, Read Memory, Modify Memory — Notes: In rare circumstances, when NULL is equivalent to the 0x0 memory address and privileged code can access it, then writing or reading memory is possible, which may lead to code execution.

## Potential Mitigations

**•** Implementation: For any pointers that could have been modified or provided from a function that can return NULL, check the pointer for NULL before use. When working with a multithreaded or otherwise asynchronous environment, ensure that proper locking APIs are used to lock before the check, and unlock when it has finished. (Effectiveness: N/A)

**•** Requirements: Select a programming language that is not susceptible to these issues. (Effectiveness: N/A)

**•** Implementation: Check the results of all functions that return a value and verify that the value is non-null before acting upon it. (Effectiveness: Moderate)

**•** Architecture and Design: Identify all variables and data stores that receive information from external sources, and apply input validation to make sure that they are only initialized to expected values. (Effectiveness: N/A)

**•** Implementation: Explicitly initialize all variables and other data stores, either during declaration or just before the first usage. (Effectiveness: N/A)

## Applicable Platforms

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

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

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

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

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

## Demonstrative Examples

**•** If an attacker provides an address that appears to be well-formed, but the address does not resolve to a hostname, then the call to gethostbyaddr() will return NULL. Since the code does not check the return value from gethostbyaddr (CWE-252), a NULL pointer dereference (CWE-476) would then occur in the call to strcpy().

**•** N/A

**•** The application assumes the URL will always be included in the intent. When the URL is not present, the call to getStringExtra() will return null, thus causing a null pointer exception when length() is called.

**•** If a user supplies a malformed request or violates the client policy, the Do method can return a nil response and a non-nil err.