# CWE Detail – CWE-62

## Description

The product, when opening a file or directory, does not sufficiently account for when the name is associated with a hard link to a target that is outside of the intended control sphere. This could allow an attacker to cause the product to operate on unauthorized files.

## Extended Description

Failure for a system to check for hard links can result in vulnerability to different types of attacks. For example, an attacker can escalate their privileges if a file used by a privileged program is replaced with a hard link to a sensitive file (e.g. /etc/passwd). When the process opens the file, the attacker can assume the privileges of that process.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2001-1494: Hard link attack, file overwrite; interesting because program checks against soft links

**•** CVE-2002-0793: Hard link and possibly symbolic link following vulnerabilities in embedded operating system allow local users to overwrite arbitrary files.

**•** CVE-2003-0578: Server creates hard links and unlinks files as root, which allows local users to gain privileges by deleting and overwriting arbitrary files.

**•** CVE-1999-0783: Operating system allows local users to conduct a denial of service by creating a hard link from a device special file to a file on an NFS file system.

**•** CVE-2004-1603: Web hosting manager follows hard links, which allows local users to read or modify arbitrary files.

**•** CVE-2004-1901: Package listing system allows local users to overwrite arbitrary files via a hard link attack on the lockfiles.

**•** CVE-2005-0342: The Finder in Mac OS X and earlier allows local users to overwrite arbitrary files and gain privileges by creating a hard link from the .DS\_Store file to an arbitrary file.

**•** CVE-2005-1111: Hard link race condition

**•** CVE-2021-21272: "Zip Slip" vulnerability in Go-based Open Container Initiative (OCI) registries product allows writing arbitrary files outside intended directory via symbolic links or hard links in a gzipped tarball.

**•** CVE-2003-1366: setuid root tool allows attackers to read secret data by replacing a temp file with a hard link to a sensitive file

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: Read Files or Directories, Modify Files or Directories — Notes:

## Potential Mitigations

**•** Architecture and Design: Follow the principle of least privilege when assigning access rights to entities in a software system. Denying access to a file can prevent an attacker from replacing that file with a link to a sensitive file. Ensure good compartmentalization in the system to provide protected areas that can be trusted. (Effectiveness: N/A)

## Applicable Platforms

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