# CWE Detail – CWE-36

## Description

The product uses external input to construct a pathname that should be within a restricted directory, but it does not properly neutralize absolute path sequences such as "/abs/path" that can resolve to a location that is outside of that directory.

## Extended Description

This allows attackers to traverse the file system to access files or directories that are outside of the restricted directory.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2022-31503: Python package constructs filenames using an unsafe os.path.join call on untrusted input, allowing absolute path traversal because os.path.join resets the pathname to an absolute path that is specified as part of the input.

**•** CVE-2002-1345: Multiple FTP clients write arbitrary files via absolute paths in server responses

**•** CVE-2001-1269: ZIP file extractor allows full path

**•** CVE-2002-1818: Path traversal using absolute pathname

**•** CVE-2002-1913: Path traversal using absolute pathname

**•** CVE-2005-2147: Path traversal using absolute pathname

**•** CVE-2000-0614: Arbitrary files may be overwritten via compressed attachments that specify absolute path names for the decompressed output.

**•** CVE-1999-1263: Mail client allows remote attackers to overwrite arbitrary files via an e-mail message containing a uuencoded attachment that specifies the full pathname for the file to be modified.

**•** CVE-2003-0753: Remote attackers can read arbitrary files via a full pathname to the target file in config parameter.

**•** CVE-2002-1525: Remote attackers can read arbitrary files via an absolute pathname.

**•** CVE-2001-0038: Remote attackers can read arbitrary files by specifying the drive letter in the requested URL.

**•** CVE-2001-0255: FTP server allows remote attackers to list arbitrary directories by using the "ls" command and including the drive letter name (e.g. C:) in the requested pathname.

**•** CVE-2001-0933: FTP server allows remote attackers to list the contents of arbitrary drives via a ls command that includes the drive letter as an argument.

**•** CVE-2002-0466: Server allows remote attackers to browse arbitrary directories via a full pathname in the arguments to certain dynamic pages.

**•** CVE-2002-1483: Remote attackers can read arbitrary files via an HTTP request whose argument is a filename of the form "C:" (Drive letter), "//absolute/path", or ".." .

**•** CVE-2004-2488: FTP server read/access arbitrary files using "C:\" filenames

**•** CVE-2001-0687: FTP server allows a remote attacker to retrieve privileged web server system information by specifying arbitrary paths in the UNC format (\\computername\sharename).

## Related Attack Patterns (CAPEC)

* CAPEC-597

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: Execute Unauthorized Code or Commands — Notes: The attacker may be able to create or overwrite critical files that are used to execute code, such as programs or libraries.

**•** Impact: Modify Files or Directories — Notes: The attacker may be able to overwrite or create critical files, such as programs, libraries, or important data. If the targeted file is used for a security mechanism, then the attacker may be able to bypass that mechanism. For example, appending a new account at the end of a password file may allow an attacker to bypass authentication.

**•** Impact: Read Files or Directories — Notes: The attacker may be able read the contents of unexpected files and expose sensitive data. If the targeted file is used for a security mechanism, then the attacker may be able to bypass that mechanism. For example, by reading a password file, the attacker could conduct brute force password guessing attacks in order to break into an account on the system.

**•** Impact: DoS: Crash, Exit, or Restart — Notes: The attacker may be able to overwrite, delete, or corrupt unexpected critical files such as programs, libraries, or important data. This may prevent the product from working at all and in the case of a protection mechanisms such as authentication, it has the potential to lockout every user of the product.

## Applicable Platforms

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

## Demonstrative Examples

**•** However, the path is not validated or modified to prevent it from containing relative or absolute path sequences before creating the File object. This allows anyone who can control the system property to determine what file is used. Ideally, the path should be resolved relative to some kind of application or user home directory.

**•** However, if the user supplies an absolute path, the os.path.join() function will discard the path to the current working directory and use only the absolute path provided. For example, if the current working directory is /home/user/documents, but the user inputs /etc/passwd, os.path.join() will use only /etc/passwd, as it is considered an absolute path. In the above scenario, this would cause the script to access and read the /etc/passwd file.